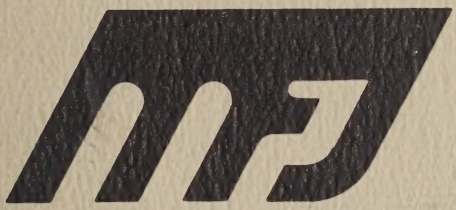


MFJ-1270 TNC 2 PACKET RADIO

OWNER'S MANUAL

CAUTION: Read All Instructions Before Operating Equipment.

MFJ ENTERPRISES, INC.

P. O. BOX 494, MISSISSIPPI STATE, MS. 39762, USA

© MFJ ENTERPRISES, INC.

• 15.838 Class B computing device: information to user.

This equipment generates and uses radio frequency energy and if not installed and used properly, that is, in strict accordance with the manufacturer's instructions, may cause interference to radio and television reception. It has been type tested and found to comply with the limits for a Class B computing device in accordance with the specifications in Subpart J of Part 15 of FCC Rules, which are designed to provide reasonable protection against such interference in a residential installation. However, there is no guarantee that interference will not occur in a particular installation. If this equipment does cause interference to radio or television reception, which can be determined by turning the equipment off and on, the user is encouraged to try to correct the interference by one or more of the following measures:

Reorient the receiving antenna

Relocate the computer with respect to the receiver

Move the computer away from the receiver

Plug the computer into a different outlet so that computer and receiver are on different branch circuits.

If necessary, the user should consult the dealer or an experienced radio/television technician for additional suggestions. The user may find the following booklet prepared by the Federal Communications Commission helpful:

"How to Identify and Resolve Radio-TV Interference Problems".

This booklet is available from the U.S. Government Printing Office, Washington, DC 20402, Stock No. 004-000-00345-4.

N6LYF

MFJ ENTERPRISES, INC.

Model MFJ-1270

Packet Radio

Terminal Node Controller

TNC 2, Rev 2

SYSTEM MANUAL

TNC 2 SOFTWARE SOURCE CODE and TNC FIRMWARE
Copyright (c) 1985
Systek
All rights reserved.

Reproduction or translation of any part of this work beyond that permitted by Sections 107 or 108 of the 1976 United States Copyright Act (or its legal successor) without the express permission of the copyright owner is unlawful except as noted below. Requests for permission to copy or for further information should be addressed to Tucson Amateur Packet Radio Corporation, P.O. Box 22888, Tucson, AZ 85734.

FIRST EDITION

1986

The information contained in this document has been carefully checked and is believed to be entirely reliable. However, no responsibility is assumed for inaccuracies. MFJ Enterprise Inc. reserves the right to make changes to any products to improve reliability, function or design without obligation to purchasers of previous equipment. MFJ Enterprises, Inc. does not assume any liability arising out of the application or use any product or circuit described herein; neither does it conve any license under its patent rights of others.

PORTIONS OF THIS DOCUMENT WERE COPIED DIRECTLY FROM TAPR DOCUMENTATION.

TABLE OF CONTENTS

CHAPTER 1 INTRODUCTION

Introduction 1-1

CHAPTER 2 COMPUTER INTERFACING

Serial Port Signals 2-1
- Apple Macintosh 2-3
- Commodore 64 or VIC-20 2-3
- IBM PCjr 2-3
- Radio Shack Color Computer 2-4
- Radio Shack Model 100 and NEC 8201 2-4
- Other Computers with 25-pin RS-232C Ports 2-4
- Other Computers with Nonstandard Serial Ports 2-5

Software Requirements 2-6
- Apple Macintosh 2-6
- Commodore 64 or VIC-20 2-6
- IBM PCjr 2-6
- Radio Shack Color Computer 2-7
- Radio Shack Model 100 and NEC 8201 2-7

Verifying Serial Port Operation 2-7
TNC 2 Serial Port Pin Functions 2-8

CHAPTER 3 RADIO INTERFACING

Method 1: Direct Connection to Microphone and Speaker . 3-2
Method 2: Accessory Jack or Interface Box Connection .. 3-4

CHAPTER 4 OPERATION I: GETTING STARTED

First Steps 4-
Entering Commands 4-
Serial Port Configuration 4-
Parity and Word Length 4-
Echos 4-
New Lines and Line Wrapping 4-
Basic Operation 4-
A Connecting and Disconnecting Exercise 4-
Digipeating 4-
Unsuccessful Connections 4-
Monitoring Channel Activity 4-
Your First Packet QSO 4-1
Starting the QSO 4-1
Digipeating 4-1
Monitoring on the Air 4-1
Special Input Characters 4-1

CHAPTER 5 OPERATION II: FURTHER DETAILS

Special Characters 5-
Operating Modes 5-
Command Mode 5-
Entering Data-Transfer Modes 5-
Converse Mode 5-
Transparent Mode 5-
Flow Control 5-
XON/XOFF Flow Control 5-
Hardware Flow Control 5-
Type-in Flow Control 5-
Packet Operation 5-1
Station Identification 5-1
Automatic Operations 5-1
Packet Formatting 5-1
Commands Affecting Protocol 5-1
Packet Timing Functions 5-1
Transmit Timing 5-1
Packet Timing 5-1
Radio Baud Rate 5-1
Special Protocol Times 5-1
Monitor Functions 5-1
Real-Time Clock and Time Stamping 5-1
HF and OSCAR 5-1
Modem Calibration 5-2
Multi-Connection Survival Guide..... 5-2

CHAPTER 6 TAPR TNC 2 COMMANDS & MESSAGES

Entering Commands 6-
Command List 6-
Messages 6-5
TNC 2 Software Release Notes Release 1.1.1..... 6-6

CHAPTER 7 HARDWARE

TNC 2 Specifications 7-1
General Description 7-3
Detailed Circuit Description 7-3
Oscillator 7-3
Dividers and Baud-rate Generator 7-4
Serial Interface 7-5
Watch-dog Timer 7-5
Modem 7-6
Power Supply 7-6
RS-232C Handshaking Protocol 7-7
Jumper Functions 7-8
Modem Disconnect - J4 7-9
Tuning Indicator Interface - J3 7-12

CHAPTER 8 TROUBLESHOOTING

General Tests 8-1
Step 1: Power Supply 8-1
Step 2: Obvious Problems 8-2
Step 3: Assembly Problems 8-2
Step 4: Cabling Problems 8-2
Specific Symptoms 8-3
Symptom: TNC appears dead 8-3
Oscillator and Reset Circuits 8-3
Digital Logic Lines 8-3
Symptom: Modem won't calibrate or key transmitter .. 8-4
Symptom: Uncopyable transmitted or received packet . 8-5
Terminal Interface Troubleshooting 8-5
Symptom: TNC won't sign on to the terminal 8-6
Symptom: The TNC appears to be signing on but 8-6
Symptom: The TNC signs on but won't accept commands. 8-7

CHAPTER 9 PACKET RADIO PROTOCOL

Explanation of Protocol 9-1
Physical Layer 9-1
Data Link Layer 9-2
HDLC Frames 9-2
AX.25 Level Two 9-3
Channel Use and Timing Functions 9-6
Channel Flow Control 9-7

BIBLIOGRAPHY

INDEX

FIGURES

Fig. 2-1. Serial port wiring for Apple Macintosh 2-3
Fig. 2-2. Serial port wiring for Radio Shack Color Computers 2-4
Fig. 2-3. Serial port wiring with jumpers for auxiliary signals 2-5

Fig. 3-1. J2 radio port connector 3-1
Fig. 3-2. Method One Interconnect 3-2
Fig. 3-3. Accessory Jack Interface 3-4
Fig. 3-4. External Interface Box 3-5

TABLES

Table 2-1. Serial port signals required by TNC 2 2-2
Table 2-2. Computers with specific serial interfacing instructions 2-2
Table 2-3. MacTerm option settings for operation with TNC 2 2-6
Table 2-4. DIP switch settings for serial port baud rates 2-7
Table 2-5. RS-232C Pin Designations 2-8

Table 4-1. DIP switch settings for 1200 baud radio data rate 4-1

Table 5-1. ASCII Codes for Control Characters 5-2
Table 5-2. DIP switch settings for radio baud rates ... 5-16
Table 5-3. Modem Calibration Parameters 5-24

Table 9-1. AX.25 Control Codes 9-5

INTRODUCTION

Thank you for purchasing the MFJ-1270 TNC 2 Packet Radio. The MFJ-1270 TNC 2 is a clone of the TAPR TNC 2. A TTL port is added making interfacing with computers which needed TTL level signals easily. The MFJ-1270 uses the latest TAPR TNC 2 software AX .25 Level 2 Version 2.0. Any future software update from TAPR will be available from MFJ Enterprises, Inc.

Your MFJ-1270 TNC 2 package contains the following items:

1. MFJ-1270
2. MFJ-1312 AC adaptor
3. System instruction manual
4. Microphone cable with 1 5-pin DIN male connector

The purpose of the MFJ-1270 TNC 2 **T**erminal **N**ode **C**ontroller (TNC) is to act as an interface between your ordinary voice radio, such as a 2-meter FM transceiver or HF SSB transceiver, and your computer. The TNC will perform all of the "magic" of establishing error-free communications between your station and another packet-radio equipped station. You will be able to have a "private channel" while sharing a frequency with other packet stations, "read the mail" of other QSOs, operate remote computer "bulletin board" or "mailbox" stations, handle message traffic -- in short, be able to enjoy all the advantages of digital communication techniques in your ham shack.

Your MFJ TNC 2 is the key to your packet station. It is based on the original TAPR TNC and inherits many of the advanced features of that design, coupled with the experience gained by thousands of TAPR-equipped Amateur packet stations worldwide.

This manual will be your guide into the realm of Amateur packet radio. Chapter 2 explains how to connect your TNC to your station computer. Chapter 3 will instruct you on interfacing the TNC to your radio and Chapters 4 and 5 will guide you through packet radio operation. Chapter 6 is a detailed breakdown of the various commands the TNC will accept and messages it may report. A description of the hardware design of TNC 2 and troubleshooting hints follow in Chapters 7 and 8. The manual concludes with Chapter 9, an overview of packet radio protocol. The Bibliography lists sources of further information on packet radio.

COMPUTER INTERFACING

You are now ready to attach your TNC 2 to your station computer or terminal. Throughout this manual we will use the term "computer" to refer to the computer or terminal you use to communicate with your TNC.

TNC 2 communicates with your computer through a serial port using signals corresponding to a standard called RS-232C. Why an RS-232C interface? Nearly every computer in production today either incorporates an RS-232C style serial port as a standard feature, or has one available as an optional accessory, either from the computer manufacturer or from a manufacturer of computer accessories.

In addition to the RS-232C port, the MFJ-1270 has a built-in TTL port to allow interfacing with computer which need TTL signals such as the Commodore 64, 128 or the VIC-20.

In order to use the TNC with your computer, the computer must have an RS-232C serial port, or a TTL serial port and a program to support the serial port. The program will typically be called a modem, terminal emulator, or communications program.

Since there are so many computers on the market today, it is impractical for this chapter to provide detailed instructions for each computer. Detailed information is given for some of the popular models available in the United States. Also provided is general computer interfacing information.

Serial Port Signals

The serial port connector on your TNC is on the rear panel and is marked "SERIAL." There are several signals available at this connector. You won't need all of them for standard packet operation. For some special applications, such as binary file transfer or some Bulletin Board operations, you may want to use more of them. In that case, see **TNC 2 Serial Port Pin Functions** at the end of this chapter.

The pins on the serial port connector of the TNC 2 that must be connected are shown in Table 2-1. Note that the TNC connects to a computer exactly as if the TNC were a standard RS-232C modem. If you have successfully used your computer

with a telephone modem, hook it up to TNC 2 in the same way. Use whatever program you ordinarily use to communicate with the modem and proceed to the section, **Verifying Serial Port Operation.**

Table 2-1. Serial port signals required by TNC 2.

Pin	Signal Name	Description
2	Transmit Data	Serial data from your computer to the TNC.
3	Receive Data	Serial data from the TNC to your computer.
7	Signal Ground	The common ground for both data lines.

If your computer is listed in Table 2-2, refer to the specific information in the following sections to connect your TNC to your computer.

Table 2-2. Computers with specific serial interfacing instructions.

Manufacturer	Model
Apple	Macintosh (tm)
Commodore	VIC-20 (tm)
	64 (tm)
IBM	PCjr (tm)
Radio Shack	Color Computer (tm)
	Color Computer 2 (tm)
	Model 100
NEC	8201

Many computers require a serial port adapter card. These cards incorporate the circuitry necessary to add an RS-232C port to the computer. Some popular models in this category are the Apple II series, the IBM Personal Computer, many Radio Shack computers, and the Sanyo MBC-55X series. If you have one of these computers with an "add-in" serial port, or if you have another computer we haven't mentioned, you should skip to one of the sections on "other computers." If your computer has a 25-pin RS-232C serial port, refer to the section Other Computers with 25-pin RS-232C Ports. Otherwise refer to the section Other Computers with Nonstandard Serial Ports.

Some computers have no serial port and no adapter is commercially available. Such computers are not suitable for use with TNC 2.

Commodore 64, C-128 and Vic-20

The MFJ-1270 TNC 2 has a built-in TTL-level port for interfacing with the Commodore 64, Vic-20 and the C-128 computers. You do not need a RS-232C converter to interface with the MFJ-1270 TNC 2.

The pin connections are indentified as follows:

MFJ-1270 J-5, Pin#	Mnemonic	Name	C-64, Vic-20, C-128* Pin ID
1	RXD	Receive Data	B,C
2	DCD	Data Carrier Detected	H
3	GND	Ground(Frame and Signal)	N
4	DTR	Data Terminal Ready	E
5	TXD	Transmit Data	M
6	DSR	Data Set Ready	L
7	CTS	Clear to Send	K
8	NC	No Connection	

*C-128 is used in the C-64 Mode for these connections.

All of these connections are not necessarily used by your terminal. The MFJ-1270 TNC 2 only needs RXD, TXD and GND connected. Programs which utilize file transfer and printer routines will probably use DTR and CTS as well. Consult your software documentation.

An optional Starter-Pack for the C-64, Vic-20 and C-128 can be purchased from MFJ Enterprises, Inc. Starter-Pack includes interface cable from TNC 2's TTL port to the user I/O port on the computer. A terminal program is also included. To order specify, MFJ-1282 for software on disk or MFJ-1283 for software on tape.

Apple Macintosh

The Macintosh serial port is an RS-422 compatible port, but it will work fine with the RS-232C serial port on your TNC. You will need a cable wired as shown in Fig. 2-1.

Fig. 2-1. Serial port wiring for Apple Macintosh.

Note that pin 1 is not connected inside the Macintosh or the TNC. If you use a shielded serial cable, which we recommend, connect both pin 1s to the shield and connect pin 1 of the TNC 2 serial connector to digital ground on the TNC circuit board. A printed circuit board pad is provided for this purpose near pin 1 of the serial connector.

IBM PCjr

The PCjr uses standard RS-232C voltage levels for its serial interface; however, the connector used is non-standard and not readily available from electronic supply dealers. Pinout information for this connector is given in the IBM PCjr Technical Reference Manual.

IBM dealers sell the "IBM PCjr Adapter Cable for Serial Devices" for converting the connector on PCjr to a standard RS-232C terminal connector. This cable attaches directly between the TNC and the PCjr. It is only about 3 inches long, however, so you may want to obtain a male-to-female RS-232C extension cable, which should be readily available.

Radio Shack Color Computer

The Color Computer series (except for the Micro Color Computer) uses a 4-pin DIN-style connector for its serial interface. Wire a cable as shown in Fig. 2-2 to interface your TNC to a Color Computer. All necessary parts should be available from Radio Shack dealers.

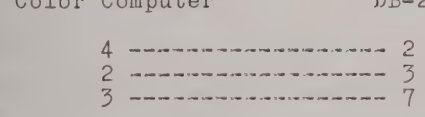


Fig. 2-2. Serial port wiring for Radio Shack Color Computers.

Radio Shack Model 100 and NEC 8201

These computers have built-in standard RS-232C serial ports that are compatible with the TNC. You will need a standard male-to-male RS-232C extension cable to connect the computer to the TNC.

Other Computers with 25-pin RS-232C Ports

If your computer has a 25-pin RS-232C port, you should consult your computer manual or accessory manual to see which pins it uses to send and receive data on, as well as which pin is used for signal common. Follow the computer manufacturer's recommendations for connecting the serial port to a modem. You may also find the technical information in this section useful.

Your TNC is configured as Data Communications Equipment (DCE), the technical term for an RS-232C modem. Most computers are configured as Data Terminal Equipment (DTE). If this is the case for your computer, you will probably be able to simply wire pin 2 of the TNC connector to pin 2 of your computer's RS-232C port, pin 3 to pin 3 and pin 7 to pin 7. You can provide these connections with a standard 3-wire male-to-female or male-to-male RS-232C extension cable, depending on whether your computer has a DB25S or DB25P connector.

If your computer is configured as DCE, you will have to wire pin 2 of your TNC to pin 3 of the computer connector, and pin 2 of the computer connector to pin 3 of your TNC. Pin 7 of the computer connector will still connect to pin 7 of your TNC serial port.
Some computers may require that pin 5 of the computer serial port connector be connected to an appropriate signal. Others may require connections for pin 8 and pin 20. You can use the computer's output signals on pins 4 and 6 as shown in Fig. 2-3.

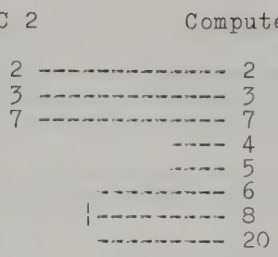


Fig. 2-3. Serial port wiring with jumpers for auxiliary signals.

Other Computers with Nonstandard Serial Ports

Computers with nonstandard serial ports must meet the following conditions.

First, the signal levels should be RS-232C compatible. The TNC requires that the voltage levels sent from the computer be greater than about +3 volts in one state and less than about +1 volt in the other state.

Second, the polarity of the signals must conform to the RS-232C standard. This means that the low voltage state must correspond to a logical "1" and the high voltage state to a logical "0".

Third, the computer must be able to correctly receive a signal which meets the RS-232C specification. The TNC supplies signals that meet this specification.

Make or buy a cable that provides the following connections. The computer serial port common pin must be tied to the TNC serial port connector pin 7. The data line that sends data from the computer must be tied to the TNC connector pin 2. The pin on which your computer receives data on must be tied to the TNC connector pin 3.

If your computer requires any other signals, you must arrange to provide them. The documentation provided with your computer or its accessory serial port should clarify any special requirements of your port.

Software Requirements

Any software package that enables your computer to act as an ASCII terminal with an ordinary telephone modem should work with your TNC. If you have a program that you have used successfully with telephone a modem and that you are familiar with, use that program to communicate with your TNC.

Apple Macintosh

Apple dealers sell a program called MacTerm for the Macintosh that works with the TNC. Load this program and set the options according to Table 2-3. In addition, set the TNC for 1200 baud as described in **Verifying Serial Port Operation**, below.

Table 2-3. MacTerm option settings for operation with TNC 2.

Compatibility	Terminal
1200 baud	VT100
7 bits/character	ANSI
even parity	UNDERLINE
Handshake Xon/Xoff	US
modem connection	80 Columns
"telephone" port	ON LINE
	AUTOREPEAT

Commodore 64 or VIC-20

A BASIC communications program is given in the Programmer's Reference Guide published by Commodore. Use the program listing for "true ASCII," as these computers use a modified ASCII format internally. You will probably want to run your TNC at 300 baud on the serial port with these computers. Setting the TNC serial port baud rate is discussed in the next section, **Verifying Serial Port Operation.**

IBM PCjr

The IBM PCjr has a built-in terminal program in the BASIC cartridge. Start this program by typing **TERM.** Refer to your PCjr BASIC manual for details on this program. For best results with PCjr, do not run the TNC serial port faster than 1200 baud. Setting the TNC serial port baud rate is discussed in the next section, **Verifying Serial Port Operation.**

Radio Shack Color Computer

There are several terminal programs available for the Color Computer. You will probably want to use a commercial program (rather than writing your own) since the Color Computer has a "software UART" that is difficult to program in BASIC.

Radio Shack Model 100 and NEC 8201

These computers have built-in terminal programs in ROM. Consult your computer's documentation for instructions in their use.

Verifying Serial Port Operation

Turn off the power to your computer and to your TNC. Connect the computer and TNC with a properly configured serial cable. Set the DIP switch on the rear panel of the TNC to the desired baud rate as shown in Table 2-4. The power **must** be OFF when these switches are set.

> **NOTE:** The serial port baud rate used between the TNC and the computer has no relationship to the baud rate used over the radio. The serial port baud rate you set on your TNC must match the baud rate used by your computer serial port.

Table 2-4. DIP switch settings for serial port baud rates.

Baud Rate	Switch 1	2	3	4	5
300	**ON**	OFF	OFF	OFF	OFF
1200	OFF	**ON**	OFF	OFF	OFF
2400	OFF	OFF	**ON**	OFF	OFF
4800	OFF	OFF	OFF	**ON**	OFF
9600	OFF	OFF	OFF	OFF	**ON**

> **CAUTION:** Only one of these switches may be ON at any time.

Turn on your computer and start the terminal program. Follow the directions for the program you are using to match the computer's baud rate with that selected on the TNC, and to set other options. Set your computer's port options to 7 bits even parity and select either 1 or 2 stop bits.

Turn on your TNC. You should see a sign-on message, which should be a readable text message, printed on your computer screen. This demonstrates the ability of your computer to accept data from your TNC. If you see nothing, switch off your TNC for a few seconds, then on again. If you still see nothing, verify your wiring and restart your terminal program. If you see gibberish on your screen you should verify that you have set the same baud rate for the TNC and the computer.

When you have successfully read the sign-on message from your TNC, type

DISPLAY

followed by a carriage return. You should see a lengthy list of items on your screen. This verifies the ability of your TNC to accept and respond to input from your computer.

Your serial interface is now working. If you are completing construction of your TNC, return to the assembly manual and continue from where you were directed to this manual.

TNC 2 Serial Port Pin Functions

This section describes the pins used on the TNC's RS-232C serial port connector. It is intended for packet operators with special applications requiring hardware handshaking. This information should not be needed by most users.

Table 2-5. RS-232C Pin Designations

Pin	Mnemonic	Name
1	FG	Frame Ground
2	TXD	Transmit Data
3	RXD	Receive Data
5	CTS	Clear To Send
6	DSR	Data Set Ready
7	SG	Signal Ground
8	DCD	Data Carrier Detect
9	---	+ 12V unregulated reference
10	---	- 12V unregulated reference
20	DTR	Data Terminal Ready

Frame Ground is provided for attachment to the chassis of the TNC and the chassis of the attached device (computer or terminal). This pin is brought out to a feedthrough on the TNC 2 PC board near pin 1 of the serial connector. It is not electrically connected anywhere else on the TNC circuit board.

Transmit Data is an input line to the TNC on which the attached device sends data.

Receive Data is an output line from the TNC on which the attached device receives data.

Clear To Send is an output from the TNC signaling the attached device to send or refrain from sending data to the TNC. This line is used for hardware flow control.

Data Set Ready is an output from the TNC telling the attached device that the TNC is operational.

Signal Ground is the common, or return, path for all signals between the TNC and the attached device.

Data Carrier Detect is an output from the TNC. As normally configured, DCD reflects the status of the CON LED: It is true when an AX.25 connection exists between your TNC and another station; it is false when no connection exists. This configuration is useful when the TNC is used with a telephone style Bulletin Board system, since the AX.25 connection, analogous to a modem signal on the telephone, indicates the presence of a user. Shorting JMP1 on the TNC 2 PC board will cause this output to always be true.

Pins 9 and 10 provide access to the TNC's unregulated +/-12 volt supplies for use by an external device. These are not intended to power an accessory, and **should not be used to source or sink more than a couple of milliamperes!**

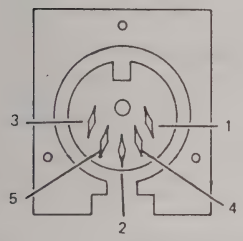
Data Terminal Ready is an input to the TNC signaling that the attached device is ready to accept data from the TNC. This line is used for hardware flow control.

RADIO INTERFACING

Computer interfacing, covered in the previous chapter, is only half the interfacing task. The other half is connecting your TNC to a radio.

Interfacing the TNC to your radio involves connecting the following signals at J2. The pinout of J2 is shown in Fig. 3-1.

Pin 1 Microphone audio, from the TNC to your transmitter.
Pin 2 Ground, audio and PTT common.
Pin 3 Push-to-talk, to allow the TNC to key your transmitter.
Pin 4 Receive audio, from your receiver to the TNC.
Pin 5 Squelch input (optional) to allow the TNC to detect activity on a shared-mode channel.



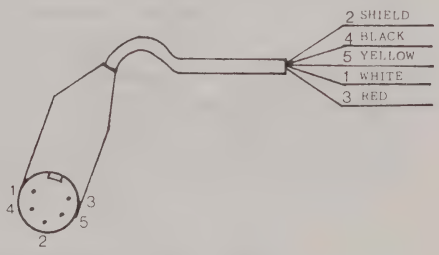


Fig.3-1.J2 radio port connector Fig.3-1a. 5-pin male DIN cable

A 5-pin male DIN connector cable is provided with your MFJ-1270. Check this cable with an ohm meter before using. Color of the wire may vary from the one shown in Fig. 3-1a.

This chapter describes how to connect these signals between your TNC and your radio and how to adjust the receive and transmit audio levels appropriately. The interconnection should be planned so as to minimize pickup of stray audio and RF noise by the lines. If possible, you should set up your packet station with a monitor speaker and be able to operate on voice without disconnecting the TNC.

This chapter assumes that you are using an FM radio. If you are operating on another mode, such as SSB, most of the information is still applicable.

Two interfacing methods are presented. You will have to use the second method if you can't adjust the audio levels properly with the first method. They require no special test equipment for adjusting the audio levels. If FM test equipment is available, however, it should be used as described. You will need a second receiver in your shack that you can use to listen to your own signal. Read the remainder of this chapter carefully before starting to interface your TNC to your radio.

Method 1: Direct Connection to Microphone and Speaker

TNC 2 was designed to allow hookup and initial testing to be done without any modifications to the radio or any signal level balancing devices in the cables.

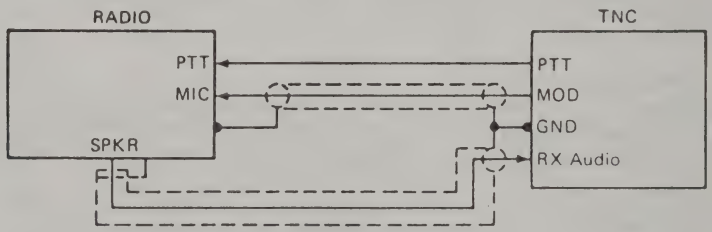


Fig. 3-2. Method One Interconnect.

For Method 1, shown in Fig. 3-2, the TNC's audio will be fed directly into the microphone connector or similarly connected auxiliary jack, and the output of the TNC will be adjusted to give a proper modulation level. The receiver audio will be taken from an earphone plug or speaker jack and fed directly to the TNC.

This method is expedient but, unless you have an auxiliary speaker/mike jack, it doesn't allow you to monitor the channel or to conveniently use the rig on voice. The connection may also be susceptible to RFI from nearby Amateur and commercial transmitters. For these reasons you may want to use the second interface method for your permanent station interface, after the initial testing phase.

Connect your TNC and radio as shown in Fig. 3-2. Turn on your TNC and computer and start your terminal program. Connect the radio to a dummy load and listen to the transmission with another nearby radio.

The transmit audio level is pre-set at the factory to be compatible with the mic input of most radios. However, if the transmit audio is too low or distorted, re-calibration may be needed. Use the following procedure to calibrate:

TRANSMIT AUDIO LEVEL ADJUSTMENT (I)

1. Enter the modem calibration procedure by typing

   CALIBRA

   and a carriage return. Press the K key on your keyboard to key the transmitter, then tap the space bar until the higher of the two tones is heard. Pressing the K key again will unkey the transmitter. After the transmitter has been keyed for a few seconds, it will be shut off automatically by the transmit watch-dog circuit. As you perform the adjustments below, you will have to periodically unkey then re-key the transmitter by typing the K key. If you wish to defeat the watch-dog timer, place JMP4.

2. With the TNC keying the transmitter and transmitting the higher of the two tones, adjust the transmit audio level as follows. With a small screwdriver, adjust trimpot R76 while you listen to the monitoring receiver. Turn the adjustment screw on R76 clockwise (CW) until no increase in output level is heard at the monitoring receiver.

3. Rotate the adjustment screw of R76 counterclockwise until the audio signal on the monitoring receiver is slightly but noticeably, reduced from the maximum level.

4. Press the K key to return to receive mode and type Q to exit the calibration routine. Be sure to remove JMP4 if you placed it to defeat the watch-dog timer. You have now set your transmitter deviation to approximately the correct level.

5. With your radio in the receive mode, open the squelch control so that a steady hiss would be heard on a speaker. Set the volume control so the DCD LED on your TNC flickers occasionally with no received signal. This is approximately the proper level for best receive performance from your TNC's modem.

If you notice a significant hum level in the monitored audio in Step 3, take measures to remove it. This may require shielded wire (recommended in any event) in your microphone audio circuit. If your transmitter has an adjustable microphone gain control, try reducing the sensitivity of the transmitter microphone circuit and increasing the signal level from your TNC to minimize hum or other noise problems.

Method 2: Accessory Jack or Interface Box Connection

If your radio has an accessory jack with PTT, transmit audio, and receive audio signals, the interface can be done through this jack with the addition of only a single resistor inside the radio (shown in Fig. 3-3).

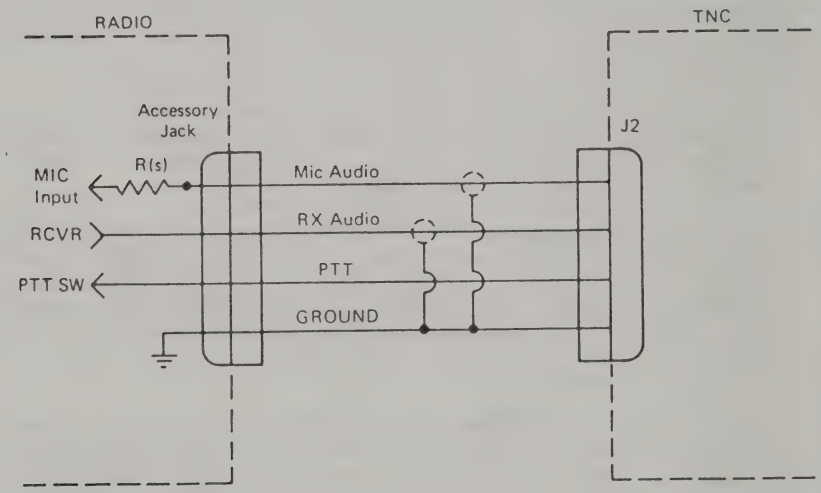


Fig. 3-3. Accessory Jack Interface.

If your radio does not have an accessory jack and you don't wish to add a connector to your radio, you should construct a separate external interface box. This box will permit simultaneous connection of your TNC and a microphone. It will either have its own speaker or pass the speaker signal through to a connector so that you can monitor the receive audio while your TNC is hooked up. A schematic of an exter-

nal interface box is shown in Fig. 3-4.

An interface box similar to the one shown in Fig. 3-4 is available from MFJ Enterprises, inc. Model No. is MFJ-1272. Price of the MFJ-1272 is $29.95.

Regardless of whether you use an accessory jack or an exter nal interface box, you should use shielded wire for all signal-carrying leads. The connector types and pinouts will be determined by the connector jacks on your radio.

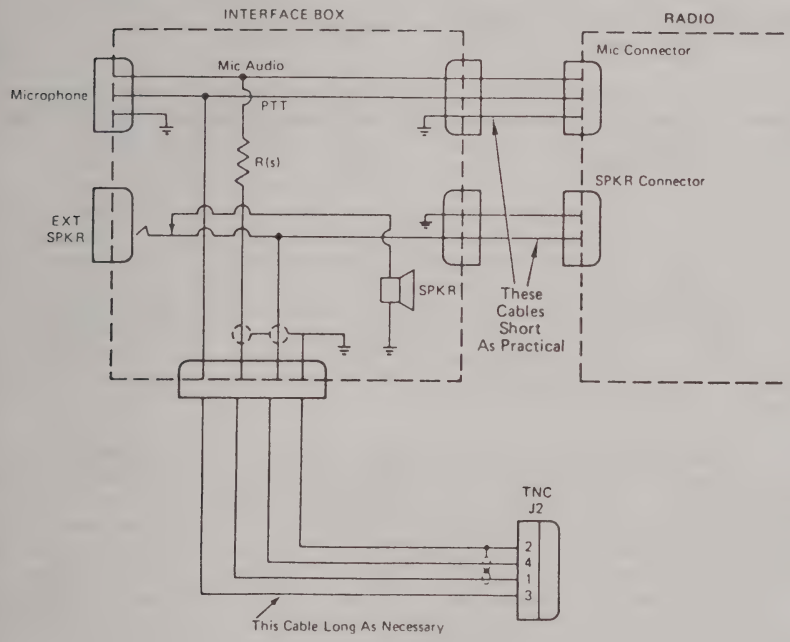


Fig. 3-4. External Interface Box.

If you built the external interface box as in Fig.3-4, then follow this procedure to adjust R(s).

1. Remove the shunt attenuator element, R57 in the micro-phone audio section of our TNC.

2. Temporarily solder a variable resistor in place of R(s) in Fig. 3-3 or Fig. 3-4. The maximum value of this resisor must be determined by experiment, but 500k should handle most cases.
3. Connect your TNC to the radio. Connect the microphone to the radio, or to the interface box if one is being used. Connect the radio to a dummy load and listen to the transmission with another nearby radio. Adjust R(s) for proper modulation.
The transmit audio level on the TNC 2 is pre-set at the factory to be compatible with the mic input of most radios. However, if transmit audio is too low or distorted, recalibration may be needed. Use the following procedure:

TRANSMIT AUDIO LEVEL ADJUSTMENT (II)

1. Turn on your TNC and computer and start your terminal program. Enter the modem calibration procedure by typing

CALIBRA

Press the K key to key the transmitter, then tap the space bar until the higher of the two tones is heard. Pressing the K key again will unkey the transmitter. After the transmitter has been keyed for a few seconds, it will be shut off automatically by the transmit watch-dog circuit. As you perform the adjustments below, you will have to periodically unkey then re-key the transmitter by typing the K key. If you wish to defeat the watch-dog timer, place JMP4.

2. With the TNC keying the transmitter and transmitting the higher of the two tones, adjust the transmit audio level as follows. With a small screwdriver, adjust trimpot R76 and set the output of your TNC to about 70% of its maximum undistorted value as observed on an oscilloscope monitoring the transmit audio at JMP7. This level may be estimated without an oscilloscope by turning R76 clockwise to maximum (indicated by a clicking sound inside the trimmer) and then backing off 15 turns. At this point the signal amplitude at JMP7 should be about 2.5 volts peak-to-peak.

3. Adjust the variable resistor installed in step 2 for proper modulation level (typically between 3.0 and 4.5 kHz deviation for Amateur FM). If FM test equipment is not available, adjust R(S) as described in steps 2 and 3 of Method 1 above, using R(S) instead of R76.

4. Press the K key to return to receive mode and type Q to exit the calibration routine. You have now set your transmitter deviation to approximately the correct level. Remove

JMP4 if you placed it in Step 4.

5. Carefully remove the variable resistor and measure its value. This is the proper value of R(S) for your particular radio.
6. Select the nearest standard value fixed resistor (1/4 watt is fine) and permanently install this resistor as R(S) in the interface circuit.
7. If you have access to FM test equipment, check to see that the modulation level is still within the limits of 3 to 4.5 kHz deviation. If it is not, make a final adjustment with the TNC transmit audio level control, R76.

8. Open the squelch control on your radio so that a steady hiss is heard. Set the volume control so the DCD LED on your TNC flickers occasionally. This is approximately the proper level for best receive performance from your TNC's modem.

NOTE: If you experienced harmonic distortion to your radio from the TNC 2, adjust capacitor trimmer (C47, next to the crystal). turn it cc or ccw until the interference disappeared.

OPERATION I: GETTING STARTED

This chapter will guide you through the basics of packet radio operation with TNC 2. Packet radio has a great deal of power and flexibility, and this chapter only scratches the surface of your packet station's capabilities. However, it contains the basic information required to get you on the air with packet radio to begin exploring this new mode.

First Steps

The serial port baud rate switches on the back of the TNC should be set to the same baud rate as your computer. These settings are described in Chapter 2, Computer Interfacing. Set the following switches, with the TNC OFF (be sure only one of these switches is ON when the TNC is powered up):

Table 4-1. DIP switch settings for 1200 baud radio data rate.

Switch	Setting
6	OFF
7	ON
8	OFF

This sets a radio baud rate of 1200 baud. The radio baud rate is not related to the computer baud rate.

Turn on your TNC. You should see something like the following display.

```
:A

MFJ ENTERPRISES, INC.
MODEL 1270.  TNC 2 PACKET RADIO
AX.25 Level 2 Version 2.0
Release 1.1.2 - 16K RAM
Checksum $CF
cmd:
```

Note: The ":A" may display as a vertical-bar and any capital letter from "A" through "J".

The first five lines are the sign-on message, which you will normally see only when you power up the TNC. (See the Software Release Notes for values of nn and xx.) The Com-

mand Mode prompt **cmd:** will appear when the TNC is in Command Mode and is ready to accept your instructions.

You may see some anomalies in the appearance of the display, which will be corrected as you proceed through the next section. The sign-on message may appear double-spaced, or characters you type may be displayed twice. You may even see incorrectly displayed characters.

Entering Commands

In the examples in this chapter, text that you are supposed to type will appear in **boldface**. Text typed by the TNC will appear in normal type:

```
cmd:RESET
```

This means that you are supposed to type the text, "**RESET**", following the Command Mode prompt which the TNC typed. End the line with a carriage return. All command entries will end with a carriage return, abbreviated **<CR>**. The <CR> at the end of a command normally won't be mentioned.

You should see the TNC's response to the **RESET** command:

```
bbRAM loaded with defaults

MFJ ENTERPRISES, INC.
MODEL 1270, TNC 2 PACKET RADIO
AX.25 Level 2 Version 2.0
Release 1.1.2 - 16K RAM
Checksum $CF
cmd:
```

The TNC has reloaded all user-settable parameters normally stored in the battery backed-up RAM (bbRAM). All parameters are now set to their default values. The TNC has also re-initialized itself exactly as it would upon power-up, and typed a sign-on message. You probably won't use this command often. The TNC will automatically reload the bbRAM at power up if it finds that the data is bad.

If you make mistakes while entering commands to the TNC, you can make corrections. To erase the last character you typed, enter a <BACKSPACE> character. The TNC will throw away the last character you typed (unless you are at the beginning of a line) and try to erase the character from your screen. Input editing will be discussed in more detail in the section, "Special Input Characters."

You are now ready to start setting up the parameters you will use. You may be satisfied with most of the defaults

for now, but a few parameters will have to be changed!

```
cmd:MYCALL WA7QST
was        NOCALL
cmd:
```

Type the text, "**MYCALL WA7QST**", following the command mode prompt. Of course, you should substitute your own call sign for **WA7QST**. Don't forget the <CR> at the end of the line. Your call sign will be used by the TNC as its "address." The TNC responds by telling you the previous value of the **MYCALL** parameter, and gives you a new Command Mode prompt.

Now try typing just the command by itself:

```
cmd:MYCALL
MYCALL     WA7QST
```

You can see the current value of most parameters by typing the command that sets the parameter followed by just a <CR>. This verifies that the TNC accepted your call sign.

The next section describes the commands you will use to configure the TNC for proper text display for your particular computer. You may not use these commands again unless you change computers or terminal programs. The following sections, "Basic Operation" and "Monitoring Channel Activity," describe the commands you will use for your everyday packet operations. With these commands you will be ready for the section, "Your First Packet QSO." The last section of the chapter, "Special Input Characters" contains information on input editing and other special characters used by the TNC.

If you intend to use your packet station for "advanced" applications, such as a computer Bulletin Board or binary file transfers, you should continue with Chapter 5, Operation II: Further Details. However, Chapter 4 contains all the information that is needed for most packet operations. For a full description of all commands, refer to the appropriate entry in Chapter 6, TAPR TNC 2 Commands.

Most commands can be abbreviated, and the minimum abbreviations for each command are given in the listings in Chapter 6. For the sake of clarity, only the full command names are used in this chapter.

Serial Port Configuration

This section describes the commands you will use to set up your TNC to work best with your computer.

Parity and Word Length

If messages from your TNC appear garbled, with incorrectly displayed characters, you may need to change the TNC's serial port parity and word length. (We assume that you have set the baud rate correctly. See Chapter 2 if the baud rate needs to be changed.) The most common parity and word length combinations are 7 bits, even parity (the TNC default), and 7 bits, space parity. The TNC default will probably be accepted even if your computer actually uses the latter setting. If your computer receives 8 bits as data, you may have to set space parity, since text may otherwise be interpreted as graphics or other special characters. To set 7 bits, space parity, use the following combination:

```
AWLEN 8          (8-bit words)
PARITY 0         (no parity bit)
```

To return to 7 bits, even parity, set

```
AWLEN 7          (7-bit words)
PARITY 3         (even parity)
```

One of these combinations will satisfy most computers. You are more likely to require a different setting if you have a terminal rather than a computer, or if you have configured your terminal port for some special application.

If your computer requires odd parity, set **PARITY 1**. If your computer detects framing errors, try setting

```
AWLEN 7          (7-bit words)
PARITY 0         (no parity bit)
```

for shorter characters. For longer characters, set:

```
AWLEN 8          (8-bit words)
PARITY 1 or PARITY 3
```

Echos

You may see two characters on your screen for every character you type, for example:

```
cmd:RREESSEETT
```

Your computer is echoing the characters you type, and the TNC is also echoing them. In this case, set **ECHO OFF** to stop the TNC's echos. If you later use your TNC with a different computer, or with a different terminal program, you may see nothing displayed when you type. In that case, set **ECHO ON**.

New Lines and Line Wrapping

If everything displayed appears to be double-spaced, your computer is adding an extra linefeed (<LF>) whenever it displays a carriage return (<CR>). Set **AUTOLF OFF** to keep the TNC from also adding an <LF>. If you change equipment you may have to set **AUTOLF ON** to restore the TNC's automatic linefeeds.

The screen-width parameter is set by default to 80, the width of many CRT displays. The TNC will send an extra <CR> (or <CR> <LF> if **AUTOLF** is **ON**) when 80 characters have been displayed on a line. If your computer does not automatically break long lines, you will need to set the screen width to the width of your display. For example, for a computer using a TV set for a display, you would set **SCREENLN 40**. If your computer does automatically break long lines, you should set **SCREENLN 0** to disable this feature on the TNC. Otherwise, you will get two <CR>s when the line wraps around.

A few computers will frequently lose the first characters of a line when several lines are typed in rapid succession, for example, in the sign-on message. You can give the computer more time between lines by setting **NUCR ON** (delay after <CR>), or **NULF ON** (delay after <LF>). The delay is adjusted by **NULLS**, which sets a number of character-times for the delay.

Basic Operation

You can learn quite a bit about your TNC's operation without actually transmitting anything. For your first experiments, your TNC will be "talking to itself," allowing you to become familiar with it before you go on the air.

Disconnect your radio from your TNC and turn off the TNC. Install the digital loopback jumper, JMP10. The analog loopback jumper, JMP7, should not be installed. Connect your computer to the TNC with your serial cable. Turn on the computer and start your terminal program.

A Connecting and Disconnecting Exercise

Packet radio QSOs are started by a connect process, which sets up the "handshaking" between the two stations that insures error-free communications. QSOs are terminated by a disconnect process, which leaves both stations free to start new QSOs. Packet QSOs can also make use of digipeaters, other packet stations which can automatically relay packets from one station to the other over a specified route.

To see how this works, you can have your TNC connect to itself. Since you have set the TNC up for digital loopback, it will receive all packets that it sends. Try the following:

```
cmd:CONNECT WA7QST
*** CONNECTED to WA7QST
```

replacing **WA7QST** with your own call sign. The TNC generates packets initiating and confirming the connection. The packets aren't actually converted to audio signals and transmitted over the radio, but they are otherwise just like packets you will be transmitting later on.

The ***** CONNECTED to** message tells you that the connection was successful. You should also notice that the CON LED has lit up and that you do not see a new **cmd:** prompt on the next line. You are now in Converse Mode, ready to start talking. Try it. Type your message, ending the line with a <CR>.

```
Hello, there.
Hello, there.
```

The <CR> causes your message to be put into a packet, or "packetized," and transmitted. (We explain in the next chapter how you can use a different character to send packets.) The underlined text is a message that the TNC received in a packet and displayed. Whenever you are in Converse Mode anything you type will be assembled into a packet addressed to the station you are talking to and transmitted. If there isn't a QSO (connection) in progress, the packet will be sent to the address **CQ**.

In the example above, your TNC entered Converse Mode automatically after the connect took place. You can also command the TNC to move back and forth between Command Mode and Converse Mode.

To return to Command Mode, you must enter a special character, Control-C (abbreviated <CTRL-C>), or else send a BREAK signal. "Control" characters are usually entered by holding down a special control key and then typing another key without releasing the control key. If your keyboard doesn't have a key marked **CTRL** or something similar, consult the documentation for your computer or terminal program to see how to enter control characters. A BREAK signal is a special transmission (not an ASCII character) which your computer may be able to produce.

> **NOTE:** If <CTRL-C> will cause your computer to do something to interfere with packet operations, such as halting the terminal program, and you can't send BREAK signals, you will have to change

the character that returns you to Command Mode. See the section on "Special Input Characters," below.

Now type a <CTRL-C>. The TNC doesn't echo the <CTRL-C>, but you should immediately see a Command Mode prompt. To return to Converse Mode, enter the command **CONVERS**:

```
<CTRL-C>
cmd:CONVERS
Whatever I type in Converse Mode is transmitted.
Whatever I type in Converse Mode is transmitted.
<CTRL-C>
cmd:
```

To terminate the QSO, you must end the connect by giving the **DISCONNE** command. The TNC will transmit packets terminating the conversation and notify you when the disconnect is complete:

```
cmd:DISCONNE
*** DISCONNECTED
```

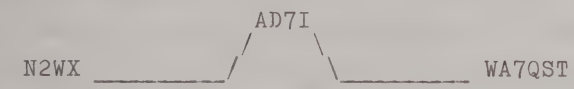
An actual QSO might be terminated by the other station, of course. In that case, you would see the ***** DISCONNECTED** message without having issued the command.

You have just performed the basic operations of any packet QSO. You established a connection with the desired station to begin the QSO, sent and received some messages, and disconnected from the station at the end of the QSO.

Digipeating

You may wish to have a QSO with another packet station that is beyond your direct radio range. If a third packet station is on the air and both you and the station you want to talk to are in range of this third station, that station can relay your packets. You set up the packet routing when you initiate the connection. Your TNC will then automatically include the routing information in the packets it sends.

The diagram below shows an example situation in which digipeating is useful.



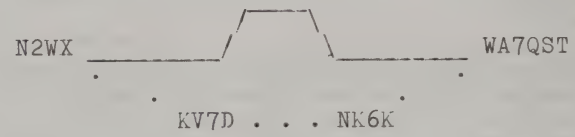
You are station WA7QST, and you want to have a packet QSO with N2WX. There is a mountain in the way and you are not in simplex range of each other. However there is a station

located on the ridge, AD7I, which is in range of both you and N2WX.

You direct the TNC to set up a connection to N2WX using AD7I as an intermediate digipeater as follows:

cmd:**CONNECT N2WX VIA AD7I**

You can specify a routing list of up to eight intermediate stations. For example, consider a modification of the example above:



AD7I has turned off his station, but you can contact N2WX by going around the mountain through NK6K and KV7D. This time you issue the connect command like this:

cmd:**CONNECT N2WX VIA NK6K, KV7D**

You specify the digipeaters in the order you would encounter them going from your station to the station to which you wish to connect.

Your station can also act as a digipeater for other stations. This doesn't require any special actions on your part -- your TNC will do everything automatically. If your station is digipeating, you may occasionally notice your transmitter keying during lulls in your own conversations.

Unsuccessful Connections

Sometimes you will initiate a connect sequence that can't be completed. The station may not be on the air, or it may not be within range of your station. You may have even mistyped the other call sign. If the TNC does not get a response to its first connect packet, it will try again. You can control the number of attempts the TNC will make with the command **RETRY**. The default number of retry attempts is 10. If the TNC doesn't get an answer after this number of transmissions, it will give up and display the message

*** retry count exceeded
*** DISCONNECTED

The retry count is also used once the QSO has started. Each transmission sent to the other station is "acknowledged," or ACKed by the other station, and vice versa. The ACK means that the packet was received and that the CRC checksum indicated that it was received without errors. This is the means by which packet radio can ensure error-free communications. Sometimes a packet won't be received correctly by the other station, either because of accidental interference from another packet station (a collision), or because of other channel noise. If your TNC doesn't get an ACK soon enough, it retransmits the packet and increments the retry count. If the count set by **RETRY** is exceeded, the TNC will automatically disconnect and display the same message:

```
*** retry count exceeded
*** DISCONNECTED
```

The automatic disconnect feature keeps a TNC from indefinitely retransmitting a packet and tying up the channel under hopeless conditions. For example, an intermediate digipeater might have been shut down, or the RF channel might have deteriorated to the point of being unusable. The other operator might have even turned off his station without disconnecting. If you are operating under special conditions, such as a marginal HF channel, you can set **RETRY 0** to disable all automatic disconnects (the retry limit is never reached).

Monitoring Channel Activity

In addition to displaying messages from the station you are connected to, your TNC can allow you to monitor other packet activity on the channel. You can "read the mail," displaying packets between other stations. Your TNC will also keep track of stations heard during a session. This section will describe some of the monitor functions.

Monitoring is enabled or disabled by the **MONITOR** command. You can try this out in digital loop-back mode while disconnected. Type:

```
cmd:MONITOR ON
cmd:CONVERS
This is a test packet.
WA7QST>CQ:This is a test packet.
```

Since you aren't connected to another station your packets are sent to the address "CQ," i.e., anyone. The packet you sent was "heard" by the TNC and displayed, along with the sending station and the destination.

If you also want to see any intermediate digipeater stations being used, you can set **MRPT ON**. This feature would be useful if you later want to connect to one of the stations you are monitoring and will need a digipeater route in order to reach it. For example, you might see the following display:

WB6YMH>WDOETZ,KV7B:Hello, Bill!

This packet was sent from WB6YMH via KV7B to WDOETZ.

If there are several digipeaters, or if the message lines are long, the display may be difficult to read. You can put the address header on a separate line from the text by setting **HEADERLN ON**:

```
WB6YMH>WDOETZ,KV7B:
Hello, Bill!
```

Ordinarily, your TNC will stop displaying monitored packets if you connect to another station, permitting you to converse without interruption. If you want to monitor activity while connected to a packet station, set **MCON ON**.

To display a list of stations heard since the last time your TNC was powered up, type

```
cmd:MHEARD
AD7I
WA7GXD
N2WX
NK6K
KV7B*
```

The last several stations whose packets were heard by your TNC are displayed. The entry "KV7B*" means that KV7B was heard digipeating a packet rather than sending one of his own. You can clear the "heard log" with the command **MHCLEAR**.

You can see the settings of the monitor parameters described above, as well as several others, by typing **DISPLAY MONITOR**.

Your First Packet QSO

Although there are still a number of features you should be familiar with for comfortable packet operation, you are probably eager to get on the air and try out your TNC. Arrange to have another packet operator get on the air to help you get started. Make sure that your friend will be close enough to ensure solid copy, with no FM "popcorn"

noise. It's best if you can get an experienced packet operator to help you get started. If you are both beginners, try to have both stations in the same room and operate on low power or into dummy loads.

Remove the digital loopback jumper, JMP10. Connect your radio to your TNC. Turn on your computer, the TNC, and your radio. Be sure you have adjusted your TNC and radio according to one of the methods described in Chapter 3. When the other station transmits, the DCD LED on your TNC should glow steadily for the duration of the transmission. You can work through the remainder of the examples in this chapter while you try out your TNC on the air.

Starting the QSO

You are ready to initiate a connect. For the sake of example, we will continue to use WA7QST in place of your call sign, and we will use WB0QRP for your friend's call. Make sure you are in Command Mode, and type

cmd:**CONNECT WB0QRP**

After a moment you should see the message

*** CONNECTED to WB0QRP

and you will be in Converse Mode. Your friend will see the message

*** CONNECTED to WA7QST

and he will also be in Converse Mode. You have begun your first QSO.

If you have trouble connecting, make sure your microphone drive level is set properly, as described in Chapter 3. It may be helpful to have an experienced packet operator listen to your transmissions and monitor with his TNC. You can also try the following procedure. Both you and your friend should set **MONITOR ON**, enter Converse Mode and send some packets. Each station should display packets sent by the other. If only one station is "hearing" properly, you can concentrate on the modulator and transmitter of that station and the demodulator and receiver of the other station. You can try experimenting with the **TXDELAY** timing parameter for the sending TNC. Set **TXDELAY 64** for a long delay. If this solves the problem, you can back off to the smallest value that works consistently.

Exchange several messages to get a feel for this new mode.

If you monitor the radio transmit indicators and listen to the speaker audio from the two rigs, you will have a better idea of what is happening. You radio will be inactive most of the time, even while you are actually typing. When you get to the end of a line and type a <CR>, your radio will be keyed briefly and your friend will hear a "brrrraaaap" on his speaker. As your message is displayed on his computer, his radio will be keyed for an even shorter time and you will hear a "brraap" on your speaker. This is the ACK, or packet acknowledgment coming back. Your TNC takes note that the packet was received correctly, but nothing is displayed on your screen.

Digipeating

Now that you are on the air, you and your friend can try out the TNC's digipeating capabilities. This is actually more interesting if you have at least three stations participating, but you can get the feel for it with two stations.

Return to Command Mode and disconnect from the other station:

```
<CTRL-C>
cmd:DISCONNE
*** DISCONNECTED
```

Now issue the following command.

```
cmd:CONNECT WA7QST VIA WBOQRP
```

As before, substitute your call for WA7QST and your friend's call for WBOQRP. You are requesting a connect to yourself, as you did before in digital loop-back mode, but this time you are using a sort of RF loop-back. You transmit packets to your friend's TNC, which relays them back to you. When the connection is established you will see

```
*** CONNECTED to WA7QST VIA WBOQRP
```

and you will be in Converse Mode. Your friend won't see anything displayed on his computer and his TNC's state won't be affected at all by your QSO. In fact, your friend could issue this connect request,

```
cmd:CONNECT WBOQRP VIA WA7QST
```

and you can carry on two separate conversations completely independently. Monitor the radio transmit indicators and listen to the speaker audio. See if you can follow the packets and the acknowledgments back and forth.

Monitoring on the Air

This is a good time to try out the TNC's monitor functions. While you and your friend are separately connected, type

```
<CTRL-C>
cmd:MONITOR ON
cmd:MCON ON
cmd:CONVERS
```

You will be able to see both your "conversation" and your friend's conversation. Also try **HEADERLN ON** and **MHEARD.**

Special Input Characters

The TNC has a number of special characters that can be used to control its actions. Many of these special characters can be used to "edit" commands and packet text as they are entered. These features can all be customized to suit you and your computer. Most of the special input characters we will describe are active in both Command Mode and Converse Mode; the exceptions will be noted.

The character used to return to Command Mode from Converse Mode is by default a <CTRL-C>. (Sending a BREAK signal also works.) This character does nothing in Command Mode, so if you accidentally enter it twice you won't mess up the next command line. You can change the Command Mode entry character with the command **COMMAND.** This is one of several commands that set special character functions. You can choose any character for this function, by entering the ASCII character code for the key. For example, you can use a <CTRL-E> to enter Command Mode by setting

```
cmd:COMMAND 5
was         $03
```

The TNC displays the previous value in hex, and you can also enter character codes in hex if you prefer. All of the special characters described below can be changed in the same way as **COMMAND.**

We have already mentioned that you can erase mis-typed characters by typing the <BACKSPACE> character. You can change this character with the command **DELETE.** If you set **DELETE ON**, you can erase characters by typing the <DELETE> character; setting **DELETE OFF** returns to using <BACKSPACE>. You will probably want to use the same key that your computer normally uses to rub out characters. <BACKSPACE> is more commonly used than <DELETE> by personal computers. If you aren't sure whether your rubout key produces <DELETE> or

<BACKSPACE> characters, you can try both settings of the **DELETE** command and see which works.

When you rub out a mis-typed character, the TNC will attempt to correct the screen display. This will work for most computers as well as display-type terminals. It won't work for hardcopy-type terminals or possibly with a few computers. If your display doesn't look right after you rub out a character, try setting **BKONDEL OFF**. The TNC will not try to correct the display but will indicate the rubout with a "\" character (<BACKSLASH>). You can restore display correction by setting **BKONDEL ON**.

If you make several mistakes in a line, or if you change your mind, you may want to cancel the whole line rather than rubbing out the characters one at a time. You can cancel the line by typing <CTRL-X>. The TNC will display a <BACKSLASH> followed by <CR>. If you are in Command Mode, you will see a new prompt:

```
     cmd:Hi , John , how are you?<CTRL-X>\
[You started typing text while in Command Mode.]
     cmd:CONVERSE
     Hi , John , how are you?
```

The cancel-line character can be changed to any ASCII character by the command **CANLINE**.

If you have changed your input by rubbing out and retyping characters, you may want to see a "fresh" copy of your input, especially if you have set **BKONDEL OFF**. The TNC will retype the line you are entering when you type <CTRL-R>:

```
     cmd:CONNECT KB7\\\WA7<CTRL-R>\
[You mis-typed the call sign.]
     cmd:CONNECT WA7GXD
```

Here the user mis-typed the first three characters of the call sign and rubbed them out. The TNC displayed "\" for each character rubbed out. The user then retyped the characters correctly and redisplayed the line. He finished typing the call sign on the new line. The redisplay-line character can be changed to any ASCII character by the command **REDISPLA**.

If your TNC displays information faster than you can read it before it scrolls off the screen, you can halt the display by typing <CTRL-S>. To resume output from the TNC to your computer, enter <CTRL-Q>. These characters can be changed to any ASCII character by the commands **STOP** and **START**, respectively.

You may occasionally want to include one of the special input characters in a packet. For example, to send several lines at once in the same packet, you would have to include <CR> in the packet at the end of each line, bypassing its "send-packet" function except at the actual end of the packet. You can include any character in a packet including all special characters by prefixing it with the pass character, <CTRL-V>. For example,

I wasn't at the meeting.<CTRL-V><CR>
What happened?

Ordinarily, this message would be sent as two packets. By prefixing the first <CR> with <CTRL-V>, the operator sends it all at once, but maintains the <CR> in the text. The pass character can be changed to any ASCII character by the command **PASS.**

OPERATION II: FURTHER DETAILS

This chapter describes some aspects of packet operation that you don't need to be concerned with for everyday conversational operation. You will want to consult this chapter if you are using your station for special applications such as a computer Bulletin Board, binary file transfers, or a "host" program. You will also find some of this material useful if you intend to operate on HF or OSCAR, or if your radio has special timing or other requirements. Even if you don't have any of these applications, you may enjoy exploring the capabilities of your TNC.

We will use the term "computer" to refer to computers or terminals. In the command examples, the TNC's prompts and other messages are shown in ordinary type, your responses are shown in **bold face**, and received packets are shown underlined. Commands and other special keywords are shown in upper case; other text entered to the TNC is shown in upper and lower case.

Special Characters

The TNC recognizes a number of special characters for input editing, flow control, and other control functions. You can change any of these special characters to customize your TNC to suit your applications, your computer, or your whim. Most of the characters are set by commands which specify the ASCII character code for the desired character. You can disable any special character feature by setting the character value to 0. Input editing characters may be disabled with no serious effects. You should use caution in disabling the flow-control or Command Mode entry characters. Also be careful not to set two special characters to the same value.

Special characters are normally set to various control characters. Control characters are entered by holding down a special control key while typing another key. For example, control-C, or <CTRL-C> is entered by holding down the control key while typing C. If your computer doesn't have a special control key, you will have to consult your computer's documentation to see how to enter these characters. If you will have difficulty entering control characters, you can change the special characters to, for example, seldom-used punctuation.

The action of each special character is described in detail under the entry in Chapter 6 for the command that sets that character.

You can enter the code for a character in either hex (base 16) or decimal notation. The TNC displays character codes in hex. A number in hex notation is indicated by beginning the number with a **$**. The "digits" of a hex number represent multiples of powers of 16. The values 10 through 15 are represented by the letters **A** through **F**, which may be upper or lower case. For example,

$1B = 1 x 16 + 11 = 27.

Tables of ASCII character codes are available in most computer manuals. A table of ASCII codes for control characters follows.

Table 5-1. ASCII Codes for Control Characters.

Dec	Hex	Control	Mnemonic	Dec	Hex	Control	Mnemonic
0	$00	<CTRL-@>	NUL	16	$10	<CTRL-P>	DLE
1	$01	<CTRL-A>	SOH	17	$11	<CTRL-Q>	DC1
2	$02	<CTRL-B>	STX	18	$12	<CTRL-R>	DC2
3	$03	<CTRL-C>	ETX	19	$13	<CTRL-S>	DC3
4	$04	<CTRL-D>	EOT	20	$14	<CTRL-T>	DC4
5	$05	<CTRL-E>	ENQ	21	$15	<CTRL-U>	NAK
6	$06	<CTRL-F>	ACK	22	$16	<CTRL-V>	SYN
7	$07	<CTRL-G>	BEL	23	$17	<CTRL-W>	ETB
8	$08	<CTRL-H>	BS	24	$18	<CTRL-X>	CAN
9	$09	<CTRL-I>	HT	25	$19	<CTRL-Y>	EM
10	$0A	<CTRL-J>	LF	26	$1A	<CTRL-Z>	SUB
11	$0B	<CTRL-K>	VT	27	$1B	<CTRL-[>	ESC
12	$0C	<CTRL-L>	FF	28	$1C	<CTRL-\>	FS
13	$0D	<CTRL-M>	CR	29	$1D	<CTRL-]>	GS
14	$0E	<CTRL-N>	SO	30	$1E	<CTRL-^>	RS
15	$0F	<CTRL-O>	SI	31	$1F	<CTRL-_>	US
				127	$7F	<DELETE>	

Operating Modes

The TNC has three operating modes. We discussed two of these modes, Command Mode and Converse Mode, in Chapter 4. The third mode, Transparent Mode, is a data-transfer mode like Converse Mode but is intended primarily for computer data interchange rather than human conversation. We describe all three of these modes below.

Command Mode

Command Mode is used to enter commands which alter the TNC's operating parameters. The other modes are entered from Command Mode. When the TNC is in Command Mode, the Command Mode prompt,

cmd:

is printed at the beginning of each input line. Note, however, that if the TNC has received and displayed packets, the prompt may have scrolled off the screen.

The TNC will be in Command Mode after a reset or power-up. After a power-off, power-on sequence, all operating parameters of the TNC are re-initialized to the parameter stored in battery backed-up RAM (bbRAM) by the resident software. After the **RESET** command is issued all operating parameters are reset to the default values stored in EPROM. The values of most parameters are stored in a permanent but easily changed form in the bbRAM memory.

The following commands set special characters which are active in Command Mode. Refer to the discussions of these commands in Chapter 6 for details on the operation of the characters in Command Mode. Also see the section on special input characters in Chapter 4.

CANLINE	Cancel current line
CANPAC	Cancel output (Command Mode function only)
DELETE	Character deletion
PASS	Insert following special character
REDISPLA	Re-display current line
START, STOP	User's flow control characters (sent to TNC)
XOFF, XON	TNC flow control characters (sent to terminal)

The following commands enable display features which are active in Command Mode. Refer to the discussions of these commands in Chapter 6 for details on the operation of these characters in Command Mode. Also see the section on terminal configuration in Chapter 4.

AUTOLF	Add <LF> after <CR> in data sent to terminal
BKONDEL	Echo after character deletion
ECHO	Automatic echo of serial port input
FLOW	Type-in flow control
LCOK	Lower case translation
NUCR	Nulls after <CR>
NULF	Nulls after <LF>
NULLS	Null count
SCREENLN	Automatic <CR> insertion

Entering Data-Transfer Modes

There are several ways to enter a data-transfer mode from Command Mode. You can type the command **CONVERS** to enter Converse Mode or the command **TRANS** to enter Transparent Mode, and the TNC will immediately enter the specified mode. The TNC will automatically enter a data-transfer mode if you are in Command Mode when a connection is completed. You can specify the data-transfer mode for automatic entry with the command **CONMODE**:

cmd:**CONMODE TRANS**

will specify Transparent Mode, and

cmd:**CONMODE CONVERS**

will return to the default choice of Converse Mode.

The timing of the automatic entry into data-transfer mode depends on whether you or the other station initiated the connection. If you receive a connect request which your TNC accepts, you will enter data-transfer mode when the TNC sends the connect acknowledgment (ACK) and types the message ***** CONNECTED TO <callsign>**.

If you initiate the connection with the **CONNECT** command, you can control the timing of the mode change with the command **NEWMODE**. If **NEWMODE** is **OFF**, the mode will change when the connect ACK is received and the ***** CONNECTED TO: <callsign>** message is typed. If **NEWMODE** is **ON**, you will enter data-transfer mode immediately, without waiting for a successful connection. Any text sent to the TNC at this point will be queued up in packets which will wait for a successful connection before being sent. If the connect attempt fails, you will be returned to Command Mode. You will also be returned automatically to Command Mode when either station disconnects and ends the QSO.

Converse Mode

The data mode used most often for ordinary QSOs is Converse Mode. In Converse Mode, the information you type is assembled by the TNC into packets and transmitted over the radio. The send-packet character causes the input to be packetized for transmission. If you type a full packet-length of characters without typing the send-packet character, your input will be packetized and transmitted anyway.

The default send-packet character is <CR>, but you can specify any character with the command **SENDPAC**. You may also choose to have the send-packet character transmitted in the packet or not. If the send-packet character is <CR> it is natural to include it in the packet as part of the text as well as interpreting it as a command. This is accomplished by setting **CR ON**. If you use some other character to force packet transmission, you may want to set **CR OFF** and inhibit transmission of the send-packet character. If you set the send-packet character to something other than <CR>, you can cancel packets of more than one line with the cancel-packet character, which is set with the command **CANPAC**. Single-line packets can be canceled with either the cancel-line character or the cancel-packet character.

To return to Command Mode from Converse Mode you must type the Command Mode entry character, or send a BREAK signal over the serial port. A BREAK is not a regular ASCII character, but it can frequently be transmitted by typing a special key on the keyboard.

A BREAK signal is a continuous mark (or 1) signal on the serial port Transmit Data line lasting approximately 0.2 second. In fact, the timing of the signal is not very important, and most serial ports will recognize a BREAK if the mark signal lasts significantly longer than the time required for a character transmission. Because of the simple nature of this signal, it is easily possible to generate a BREAK with circuitry external to the computer, thus guaranteeing entry to Command Mode in automatic station operation.

The following commands set special characters which are active in Converse Mode. Refer to the discussions of these commands for details on the operation of the characters in Converse Mode.

CANLINE	cancel current line
CANPAC	cancel current packet
COMMAND	Command Mode entry
DELETE	character deletion
MFILTER	characters to be filtered in monitored packets
PASS	insert following special character
REDISPLA	re-display current line
SENDPAC	send current packet
START , STOP	user's flow control characters (sent to TNC)
XOFF , XON	TNC flow control characters (sent to terminal)

The following commands enable display features which are active in Converse Mode. Refer to the discussions of these commands for details on the operation of these characters in Converse Mode.

8BITCONV	Retain high-order bit from serial port in converse mode
AUTOLF	Add <LF> after <CR>
BKONDEL	Echo after character deletion
ECHO	Automatic echo of serial input
ESCAPE	<ESCAPE> translation
FLOW	Type-in flow control
LCOK	Lower case translation
NUCR	Null characters after <CR>
NULF	Null characters after <LF>
NULLS	Null count
SCREENLN	Automatic <CR> insertion

Transparent Mode

Packet radio is very well suited to transfer of data between computers. In some cases Converse Mode will work well for computer data transfer. However, files such as a .CMD file on a CP/M system, a BASIC program, or even a word-processor text file, may contain characters which conflict with special characters in Converse Mode. Some of these files may utilize all eight bits of each byte rather than the seven bits required by ASCII codes. If you transfer such files you will have to use Transparent Mode.

Transparent Mode is a data-transfer mode like Converse Mode. In this mode there are no special characters -- everything you type (or everything your computer sends to the TNC) is sent over the radio exactly as it was received by the TNC. There are no input editing features and there is no send-packet character. Packets are sent at regular time intervals or when a full packet of information is ready. The

time interval at which data is packetized is set by the **PACTIME** command.

The display characteristics of the TNC are also modified in Transparent Mode. Data is sent to the computer exactly as it is received over the radio, including all 8 bits of each byte received. Features such as auto-linefeed insertion and screen wrap are disabled, and echoing of input characters is disabled. The parameters that control these features in Command Mode and Converse Mode are not changed by entering Transparent Mode, and all display features are re-enabled when the TNC is returned to Command Mode. Most of the link status messages that appear as the TNC moves between disconnected and connected states are also disabled in Transparent Mode.

In order to permit the Command Mode entry character to be transmitted freely in Transparent Mode, the escape to Command Mode from Transparent Mode has been made a little more complicated. You can still return to Command mode by transmitting a BREAK signal, just as in Converse Mode. You can also utilize the Command Mode entry character in the following way.

You must wait for a time period after typing the last character to be sent. This time is set by the command **CMDTIME**. Following this wait, you must type three Command Mode entry characters (default <CTRL-C>) within an interval **CMDTIME** of each other. After a final **CMDTIME** interval in which no characters are typed, you will see the

cmd:

prompt. If any characters are typed during this interval (even Command Mode entry characters) the escape will be aborted and all the Command Mode entry characters that have been typed will be sent as packet data. If you set **CMDTIME** to zero you will not be able to escape from Transparent Mode using this second procedure.

Flow Control

Whenever data is transferred to computers (home computers or TNCs), there is a chance that the data will be received faster than the computer can handle it. Some programs try to deal with this by providing data buffers for storing incoming data until the program is ready for it. However, this merely postpones the problem, since there is a limited amount of room in any buffer. In order to prevent loss of data the computer must be able to make whatever is sending data stop sending, and later tell it to resume sending. If you are a home computer user, you are probably already

familiar with one type of flow control, which allows you to stop the output from the computer while you read it and restart it when you are ready for more.

The TNC's input buffer may fill up in Command Mode if you try to type too long a command. In Converse Mode the buffer may fill up for any of several reasons: you may be using a faster serial port baud rate than the radio data rate; radio data transmission may have slowed down because of noise or other users on the channel; the person or computer at the other end may have stopped output from that TNC. The TNC will signal the computer to stop sending data when there is room remaining for about 80 characters in the buffer. When the buffer fills up entirely, data will be lost. When the buffer empties so that there is room for at least 270 characters, the TNC will signal the computer to start sending data again.

A computer file transfer program may be unable to process data fast enough to keep up with output from the TNC. In order to be sure of reading every character, a computer must respond to interrupts from its I/O devices. Some simple programs may poll the input register for new data. If the polling is not done often enough, data may be lost. Some computers disable interrupts during disk accesses. If the program enters a routine which will not allow it to check for data or respond to it, it should signal the TNC to stop sending data.

There are two methods of providing flow control which are supported by the TNC. XON/XOFF flow control, sometimes called "software flow control," is accomplished by sending a special character (usually <CTRL-S>) to request that the output stop and another special character (usually <CTRL-Q>) to restart output. Hardware flow control may be used if both computers use the Clear To Send (CTS) and Data Terminal Ready (DTR) lines of the RS-232C standard.

Some commonly used terminal programs and file transfer programs for home computers do not implement flow control in software, and many serial ports do not support hardware flow control. Although the DTR and CTS lines appear at the connector, they may not be used on some computers unless the software reads the state of the CTS line. If you find that the TNC seems to lose data during file transfers, you should immediately suspect a flow control problem.

XON/XOFF Flow Control

If you are using a terminal (rather than a computer) or if your computer does not support DTR/CTS flow control, you should use XON/XOFF flow control, which is enabled by set-

ting **XFLOW ON**. The special flow control characters are set to <CTRL-S> and <CTRL-Q> by default, but they may be changed. The commands **XON** and **XOFF** set the characters which will be sent to the terminal by the TNC, and the commands **START** and **STOP** set the characters to be sent to the TNC by the terminal. Your computer may receive as many as 4 characters from the TNC after sending a STOP character, since some characters may already be "in route" through serial I/O chips.

If you send a STOP (START) character to the TNC when it is already stopped (started), the character will be ignored. If the STOP and START character are the same character, this character will "toggle" the output, turning it off if it is on, and on if it is off.

You can disable XON/XOFF flow control in one direction only by setting the appropriate flow control characters to 0. If you do this, the TNC will automatically use CTS flow control to stop input from the terminal. XON/XOFF flow control is normally disabled in Transparent Mode, since all characters are treated as data. If you cannot use DTR/CTS flow control, you may enable the XON and XOFF characters (the commands from the TNC to the terminal) by setting **TXFLOW ON** and **XFLOW ON**. The START and STOP characters (the commands to the TNC from the terminal) can be enabled in Transparent Mode by setting **TRFLOW ON**. Note that the mode is no longer truly transparent when these features are enabled.

Hardware Flow Control

Hardware flow control is less likely to depend on the programming of a particular communications program. DTR and CTS are normally used for flow control signals in Transparent Mode. The command **XFLOW OFF** enables hardware flow control in Converse Mode and Command Mode. Your computer may receive as many as 2 characters after it signals the TNC to stop sending, since some characters may already be "in route" through serial I/O chips. Refer to Chapter 7, Hardware for details on the interface required for hardware flow control.

Type-in Flow Control

Type-in flow control, enabled with the command **FLOW**, is really a display feature. It can keep the TNC from interrupting you with incoming packets when you are in the middle of typing a command line or an outgoing packet. As soon as you type the first character of a line, the TNC will put a "hold" on all output (except for echoing your input). The "hold" remains in effect until you type a <CR> to end the

command line, or a send-packet character to mark the end of a packet, or until you erase or re-display the line you have started.

Some computers have difficulty simultaneously sending and receiving characters over the serial port. This is most commonly the case for computers with "software UARTs." Type-in flow control will improve the operation of such computers with the TNC.

Packet Operation

The previous chapter's discussion of "Basic Operation" contains enough information for most packet operation. This section describes a few other aspects of packet operation.

Station Identification

Your station identification (call sign) is set with the command **MYCALL**, as described in the previous chapter. If you will have more than one station on the air operating with the same call sign, they must be distinguished -- no two stations can have identical station identifications, or the packet protocol will fail. You can distinguish additional stations by setting the "secondary station ID", or SSID. This is a number from 0 to 15, appended to the call sign with a dash:

cmd:**MYCALL W3IWI-3**

If you don't specify the SSID extension, it will be 0, and the TNC won't explicitly show SSIDs that are 0. If you want to connect to a station with a SSID other than 0, or use such a station as a digipeater, you must specify the SSID:

cmd:**CONNECT AD7I-2**

or

cmd:**CONNECT WA7GXD VIA N7CL-5**

The TNC can send an automatic identification packet every 9-1/2 minutes when your station is operating as a digipeater. You can enable this feature with the command **HID ON**. An ID packet is displayed as follows by a monitoring station:

W3IWI-3>ID:W3IWI/R

You can request a final identification as you take your station off the air with the command **ID**. The TNC will only send identification packets if it has been digipeating.

Automatic Operations

Normally, any packet station can be used by other stations for relaying, or digipeating, packets to a more remote destination. If you don't want your station digipeating packets, you can give the command **DIGIPEAT OFF**. Unless there are special circumstances, such as a station operating on emergency power, most packet operators set **DIGIPEAT ON** in the spirit of Amateur cooperation.

Your station will normally accept a connect request from another station if it isn't already connected. You can disable this capability by setting **CONOK OFF**. If you receive a connect request when **CONOK** is **OFF**, the TNC will display the message

*** connect request: <callsign>

and send a "busy signal" rejection packet to the other station. If you receive a rejection packet from a station you try to connect to, your TNC will display

*** <callsign> busy
*** DISCONNECTED

If you want to have a special message sent automatically to stations connecting to you, you can specify the message with the command **CTEXT**. This message can consist of any text string up to 120 characters, and you may include <CR>s by prefixing them with the pass character:

cmd:**CTEXT Sorry, I can't talk right now.<CTRL-V><CR>**
I'll be on the air again after 8 PM.<CTRL-V><CR>
Joe

In order for this message to be sent to stations connecting to you, you must set **CONOK ON** so that the connection takes place (default), and enable the automatic message with **CMSG ON**.

If you want to leave you station on but inhibit transmitting, you can set **XMITOK OFF**. If you do this, you would normally set **CONOK OFF** as well.

You can have your station periodically send an automatic message by enabling "beacons." A beacon can be used to make general-interest announcements, provide packets for other stations to use to test their ability to receive, or announce the presence of a bulletin-board operation. The beacon message is set with the command **BTEXT**, which works the same way as the **CTEXT** command. You enable beacon transmission and set the frequency at which beacons are sent with

the command **BEACON**. To transmit the beacon at 10-second intervals, for example, give the command

cmd:**BEACON EVERY 1**

The beacon function also has a transmit-after mode, enabled by using the keyword **AFTER** in place of **EVERY**, in which a beacon packet is only transmitted after activity is heard on the channel. This feature might be used to leave an announcement for other packet users. If someone transmits on an otherwise idle channel, a beacon can be sent a short time later. No beacons are sent in this mode if there is a lot of packet activity on the channel, since the required period of quiet will not occur.

Packet Formatting

The maximum length of a packet is determined by the command **PACLEN**. If you type more than the maximum number of characters without entering a send-packet character, the TNC will transmit a maximum-length packet. In Transparent Mode, a packet will be sent if the maximum number of characters is entered before the delay conditions set by **PACTIME** force a packet to be sent. Some TNCs may not be able to accept packets longer than 128 characters.

If you have set the send-packet character to <CR>, you probably want the <CR> to be included in the packet for display at the other end. If you set the send-packet character to a special non-printing character, you probably want the character to be treated as a command only. The command **CR** controls whether the send-packet character is to be echoed and included in the packet.

You can add a <LF> after each <CR> included in your packets by setting **LFADD ON**. If the other station reports that lines are overprinted on his display, and he can't remedy the situation at his end, you can enable this function.

Commands Affecting Protocol

This section describes some of the commands that affect the operation of the packet protocol. Details of the protocol are given in Chapter 9.

The TNC implements AX.25 Level 2 protocol, a set of rules for formatting messages to other TNCs. The version of AX.25 Level 2 protocol used by the TNC can be set to Version 2.0 with the command **AX25L2V2 ON**, or to Version 1.0 with the command **AX25L2V2 OFF**. Digipeating may not be successful if some TNCs are running Version 1.0 and some are running

Version 2.0. In addition, the command **CHECK** controls a timing function that depends on the protocol version selected.

You can specify the "address" to be used for unconnected packets, as well as intermediate digipeaters with the **UNPROTO** command. The format is similar to that of the **CONNECT** command:

cmd:**UNPROTO QST VIA NK6K**

The default address for unconnected packets is **CQ**.

The following functions may be useful for tracking down protocol problems. They are seldom useful for ordinary packet operations. The error-checking function of the protocol is disabled for monitored packets with the command **PASSALL**. If you set **PASSALL ON**, any "packet" will be displayed if it meets the following conditions: It must start with a flag sequence; and it must contain an integral number of 8-bit bytes. The **TRACE** command enables the display of the address and control fields of packets, as well as the text. The trace function displays all bytes in hex as well as ASCII equivalents.

Packet Timing Functions

Transmit Timing

Amateur radio equipment varies greatly in the time delays required in switching from receive to transmit and from transmit to receive. If the TNC starts sending data before the transmitter is operating or before the receiver has had time to switch from transmitting and lock up on the incoming signal, the packet will not be received properly. The delay between transmitter keyup and the beginning of data transmission is controlled by the command **TXDELAY**. During the time the TNC is keying the transmitter but not sending data, it will transmit a synchronizing signal (flags).

If you are transmitting packets through an audio repeater, you may require a considerably greater keyup delay than is required for direct communications. Furthermore, the extra keyup delay is not required if the repeater has not had time to "drop" since the last transmission. The command **AXDELAY** allows you to specify an additional keyup delay to allow the repeater receiver and transmitter to lock up. The command **AXHANG** sets the time the TNC will assume is required for the repeater to drop. If the TNC has detected channel activity recently enough that the repeater transmitter should still be on, it will wait only the **TXDELAY** time before sending data, rather than adding an **AXDELAY** time as well.

The commands **TXDELAY**, **AXDELAY**, and **AXHANG** all set times in units of 10 ms. If **AXDELAY** is in effect the total keyup delay will be

Keyup delay = (TXDELAY + AXDELAY) * 10

in milliseconds. If channel activity has been heard more recently than AXHANG*10 ms ago, the keyup delay will only be

Keyup delay = TXDELAY*10

in milliseconds.

Packet Timing

The AX.25 protocol provides for retransmitting packets if no acknowledgment is heard from the intended destination station within a certain period of time. A packet might not be acknowledged due to channel noise or "collision" with another packet transmission, and since there may be other stations on the channel, the receiving station may not be able to acknowledge the received packet immediately. The time lapse before the originating station retransmits the packet is set by the command **FRACK** (frame acknowledge time). The maximum number of retransmissions before the originating station terminates the connection is set by the command **RETRY**. The maximum number of transmissions of a packet is RETRY+1, since the initial transmission does not count as a retransmission. Setting **RETRY** to 0 specifies an infinite number of retries.

The frame-acknowledge time is automatically corrected for the additional time required for digipeating. The time interval before TNC 2 retransmits an unacknowledged packet is

Retry interval = FRACK * (2*n + 1)

in seconds, where n is the number of digipeaters in use for this connection.

Acknowledgments of digipeated packets are made from end to end, and digipeaters do not acknowledge the packets they relay. If there are several intermediate relays, the chance of either the original packet or the acknowledgment to be lost increases drastically. To reduce this problem, an automatic wait time can be imposed on any station not transmitting a digipeated packet. Stations waiting for a clear channel to transmit packets wait for this time interval after the channel clears before transmitting. This wait does not apply if the station will be transmitting one or more digipeated packets. This usually gives the digipeater a clear chance at the channel.

The wait time is set by the command **DWAIT**, which specifies 40 ms intervals. If no digipeating is being done by anyone in the local area, this parameter can be set to 0, but in any event it should be set to the same value by all members of a local packet group.

In order to avoid unnecessary packet retries, the TNC implements a collision-avoidance strategy which applies to all packets except those being digipeated. On the second and subsequent transmissions of a particular packet, the TNC waits an additional random time after detecting a clear channel before beginning transmission. This prevents repeated collisions of transmissions by the same two stations. The random time is a multiple (0-15) of the TXDELAY time.

The interval, in milliseconds, between the TNC detecting carrier-drop and beginning to transmit is

Wait time = DWAIT * 10

for the first transmission of a packet. For subsequent transmissions of the same packet the interval is

Wait time = DWAIT * 10 + (r * TXDELAY) * 10

where r is a random number from 0 to 15. Thus, if your TNC is forced to re-transmit packets, you will occasionally hear a fairly long delay before transmission begins.

Multiple packets may be transmitted before waiting for an acknowledgment. This permits more efficient channel use when large amounts of data are being transferred. The maximum number of packets which the TNC will send before waiting for acknowledgment is specified by the command **MAXFRAME**. This does not mean that the TNC will wait until several packets have been entered before transmitting. **MAXFRAME** in combination with the command **PACLEN**, which sets the maximum number of characters in a packet, determines how much information can be sent in a single transmission. The best combination for efficient data transfer is determined partly by the channel quality and partly by the rate at which the terminal can process data. For a 1200 baud terminal data rate, you should start with a combination that produces about 300 characters outstanding at one time.

Radio Baud Rate

The radio data transmission rate is set by switches 6, 7 and 8 of DIP switch SW2 (Do not change these switches while power is on, and make sure only one of these three switches is ON at any time while the TNC is powered). The rates available are:

Table 5-2. DIP switch settings for radio baud rates.

Radio Data Rate (baud)	SW2 Switch Setting 6	7	8
300	**ON**	OFF	OFF
1200	OFF	**ON**	OFF
9600	OFF	OFF	**ON**

Note that there is no relationship between terminal baud rate and radio baud rate. In order to communicate with another packet station you must use the same radio baud rates. The length of time required to send a given amount of information increases as the baud rate decreases. For example, it takes four times as long to send data at 300 baud as at 1200 baud. If you use slow radio baud rates, you should limit the length of transmissions by setting **MAXFRAME** to 1 and choosing **PACLEN** so that the hardware watchdog timer does not disrupt your transmissions and channel traffic is broken up frequently.

The Bell-202 compatible modem is optimized for a 1200 baud radio data rate. For HF operation at low baud rates, you should reconfigure the modem as described in Chapter 7. The on-board modem is not useful at rates higher than 1200 baud. The TNC can provide data signals at 9600 baud, but an external modem is required for such operation.

Special Protocol Times

You can set up a connection timeout with the command **CHECK**, which specifies a time in multiples of 10-second intervals. This function prevents your TNC from getting stuck in a connection when the other station disappears for longer than the specified time. The TNC uses this time somewhat differently depending on the setting of **AX25L2V2**.

The command **RESPTIME** sets a delay between the receipt of a packet and the transmission of the acknowledgment packet. This delay is used to prevent collision between an acknowledgment and another packet from the sending station. This is primarily necessary during file transfers; otherwise the delay is best set to 0. During file transfers the stations receiving the file should set **RESPTIME** to 10 or 12 (default).

The timing of packet transmission in Transparent Mode is determined by the command **PACTIME**. You can choose the way packet transmission is timed. If you are typing input to a remote computer it is usually best to have packets transmit-

ted at regular intervals. If your computer is operating a remote-host or bulletin board program you should send packets after an interval with no further input from the computer. You can enable the use of **PACTIME** in Converse Mode with the command **CPACTIME.**

Monitor Functions

The TNC's protocol is designed for setting up "circuits" between two stations. However it can also operate in a mode more suitable for a "net" or "round-table" discussion with several participants, although reliable reception of all transmissions by every station cannot be guaranteed. This is done by enabling the monitor functions. Most of the monitor functions are described in Chapter 4.

Monitoring is enabled by the command **MONITOR ON**, and separate monitor functions are individually enabled.

If connected packet QSOs are taking place on the frequency of your group conversation, you may wish to ignore all connected packets while your group operates in unconnected mode. The command **MALL OFF** causes the TNC to ignore connected packets.

If you want to be able to monitor packet activity when your station is not connected, but have all monitoring automatically cease when you connect to someone, set **MCON OFF.**

If you want to monitor stations selectively, you can set up a list of up to eight callsigns with the command **LCALLS.** The callsigns in this list are regarded as "buddies," i.e., the only stations you want to listen to if **BUDLIST** is **ON.** Otherwise, the stations in the list will be ignored, and all other stations will be monitored.

You can operate a group conversation with some data integrity by having the stations connect in pairs and setting **MALL ON** and **MCON ON.** This does not insure that every packet is received at every station, but it does insure that a packet involved in a collision will be retried. If you have an odd number of stations participating in this sort of conversation, one station can connect to himself via another station as digipeater.

For example, WB6YMH, WD0ETZ, WA0TTW, W1BEL, and K9NG wish to carry on a group conversation. In order to make all the transmissions as reliable as possible, the following connections are made.

WB6YMH connects to W1BEL
WAOTTW connects to K9NG
WDOETZ connects to WDOETZ via W1BEL

If each station specifies **MCON ON** and **MALL ON**, each station will see the packets sent by all the others.

Ordinarily, only text packets are displayed. If you want to see some of the protocol packets, you can set **MCOM ON** and connect, disconnect, disconnect acknowledgments, and DM (connect-request rejection) packets will be displayed. For example,

WB9FLW>N7CL <C>
KV7B>N2WX <D>

indicate connect and disconnect packets. Disconnect acknowledgments are designated **<UA>**, and DM packets are **<DM>**.

You can cause the TNC to "filter" certain characters from monitored packets with the command **MFILTER**. This allows you to remove, for example, form-feeds, bell characters, or extra <LF>s that may be necessary to the stations involved in a connection, but which may interfere with your display. You can specify up to four characters by giving the ASCII character codes in hex or decimal.

Real-Time Clock and Time Stamping

You can enable the TNC's real-time clock by setting the date and time with the command **DAYTIME**. Once you have set the clock, you can request the time by entering **DAYTIME** with no parameters. The format of date and time display is controlled by the command **DAYUSA**. If the TNC is powered off, **DAYTIME** will have to be reset.

Monitored packets can be time-stamped if **DAYTIME** has been set. To enable this function, set **MSTAMP ON**. You can also time-stamp connect and disconnect messages with the command **CONSTAMP ON**.

HF and OSCAR

The TAPR TNC 2 is optimized for a local VHF FM environment: The modem is configured for best response at 1200 baud. The settings of **MAXFRAME** and **PACLEN** provide the possibility of several continuous frames of long data length.

The requirements for optimum performance with a typical HF or OSCAR 10 path are very different. Lower signal to noise ratios require lower baud rates, noise spikes and fades

require shorter packet lengths, and a higher rate of false carrier detects lowers the total usable dynamic range in the audio input. The TNC hardware and software can be reconfigured to improve throughput in these environments. For HF operation at 300 baud, we recommend setting **MAXFRAME** to 1.

The TNC detects a busy channel by monitoring the lock-detect signal from the demodulator. The presence of a lock-detect signal is indicated by the Data Carrier Detect (DCD) LED. Each time DCD goes off the TNC will start a **DWAIT** interval which must elapse before the channel is considered to be available. On a noisy channel spurious lock-detect signals may be generated. For HF and OSCAR operation you should set **DWAIT** to 0. The random wait before retry transmissions can be disabled by setting **TXDELAY 0** and using **AXDELAY** to set the required keyup delay. Of course, **AXHANG** should be 0 for this application.

If you are operating a full-duplex radio station (simultaneous transmit and receive) such as an OSCAR 10 station, you should set **FULLDUP ON.** The TNC is always electrically capable of full duplex operation, but this parameter causes the protocol to behave differently in acknowledging packets. In addition, the TNC will ignore the state of the DCD line. You may be able to improve operation somewhat by disconnecting the DCD line at the modem connector (J4 pins 1-2).

Although intuition tells you that lower baud rates will reduce the number of packet retries, there is usually a small range between "too fast" and "too slow." A slower packet takes longer to transmit and is therefore a larger target for fades and static crashes. The entire packet must be received correctly in order to be accepted. Data rates of 1200 baud have been used on both HF and through OSCAR 10.

You can improve the response of the modem for low baud rates by calibrating the modulator and demodulator tones for a narrow shift, as described in the next section, and modifying the demodulator. This also permits the use of a narrow filter by the receiving station. You should set the demodulator center frequency so that it is midway between the high and low tones when the packet tones are centered in your passband. The modem parameters and timing constants are determined by the parts on headers U15 and U19. Consult the EXAR notes listed in the Bibliography for the procedure for determining component values. If you are dedicating your TNC to 300 baud operation, consider increasing the time-constant of the filter at U20 pin 3 by a factor of 4 (increase the capacitor value, not the resistor).

The audio frequency response of the TNC in combination with a SSB transceiver may be less than optimum. If you intend to operate your TNC on HF or OSCAR, you can modify the input

filtering provided by the MF-10 switched capacitor filter by changing the values of the components on header U17. Alternate filter configurations are listed from time to time in TAPR's newsletter, PSR.

The on-board modem can be completely bypassed at connector J4. You can supply an external modem which uses other modulation methods or higher baud rates. The interfaces available on J4 are TTL levels. Refer to Chapter 7 for more information.

Modem Calibration

The TAPR TNC 2 contains a versatile, user configurable FSK modem that is suitable for both 1200 baud packet use as well as lower data rate use, such as 200 Hz shift 300 baud use on HF links. Calibration of the modem is usually only required when the unit is initially put into operation or when you want to change the modem characteristics. You may want to check the calibration if you have difficulty connecting to other packet stations, but recalibration is normally not necessary.

Calibration of the TNC on-board modem is straightforward. First, determine the data rate at which you wish to operate. The required shift may then be determined, followed by the actual FSK frequencies to use. While you may select from any tones in the normal audio range, there are a few simple rules that may help.

1. The TNC only supports 300 baud and 1200 baud operation of the radio link using the on-board modem. While the modem is capable of operating at other data rates, the TNC baud-rate generator circuit only supplies clocks for 300, 1200 and 9600 baud (9600 baud is for use with an external modem).

2. Most 1200 baud operation uses 1000 Hz shift and tones of 1200 Hz and 2200 Hz. These are the default tones for which the TNC modem is optimized, using the components supplied in the kit.

3. Most HF operation below 28 MHz uses 300 baud, 200 Hz shift FSK. TAPR recommends 1600 Hz and 1800 Hz as the modulator tone frequencies, but any pair of frequencies with this separation, modulating an SSB transmitter, will work. If your demodulation center frequency is 1700 Hz, your transmissions will be readable whenever you are able to copy the other station, regardless of his actual modulation frequencies. Incidentally, the sideband may be chosen at will due to the nature of the data, although most stations conventionally use LSB.

In the example below we will set up the modem for normal 1200 baud operation.

The software calibration routine compares the frequency being calibrated against a reference signal. The desired frequency is specified by setting a count, **n**, using the command **CALSET**.

For the modulator, determine the calibration count according to the following formula:

n = (525,000 / **f**) + 1

where **f** is the desired frequency in Hertz. Substituting 1200 and 2200 Hz, we end up with:

n = 438 (for 1200 Hz)
n = 239 (for 2200 Hz)

To set the PLL demodulator, we use a different formula:

n = (262,500 / **f**) + 1

due to the frequency doubling effect of the calibration waveshaping circuitry on the unusual waveform produced by the XR2211 PLL demodulator IC.

The demodulator will typically be set to the average of the two tones of the modulator. In this case we find

(1200 + 2200) / 2 = 1700 Hz.

It turns out that experiments indicate that the somewhat lower frequency of 1685 Hz produces slightly better results than 1700 Hz (only for this particular tone combination!). Using this frequency we find

n = 157 (for 1685 Hz)

The next step is to preset the trimpots for the modulator. When adjusting the trimpots, you should hear a soft "click" when the element is adjusted to the end of its travel. Turning the element past this point will not result in damage to the trimpot; however, it is suggested that you do not make a practice of turning the adjustment screw past this point too much. Use a small-tipped screwdriver or special trimpot adjusting tool when adjusting the trimpots.

Preset R77 20 turns counter-clockwise (CCW) or until you hear the element "click" at full CCW rotation. R77 controls the higher of the two modulator output tones and this sets it to its maximum frequency.

Preset R78 20 turns clockwise (CW) or until you hear the element "click" at full CW rotation. R78 controls the lower of the two modulator output tones and this sets it to its lowest frequency.

Turn off the power to the TNC and remove it from its cabinet. To protect the battery backed-up RAM, be sure to set the TNC on a non-conducting surface -- a sheet of paper will do. Attach your terminal and follow the standard power-up procedure. When you see the command mode prompt, enter

cmd:**CALSET 438**

or use the value appropriate for the lower modulator tone. Next, enter

cmd:**CALIBRA**

This starts the software calibration routine. The "commands" to the calibration routine are single characters, and require no carriage return.

Install a push-on jumper at JMP9 pins 1 and 6 (the top two pins of JMP9) and another push-on jumper at JMP4.

> **NOTE:** During calibration the STA and CON LEDs are used as a frequency indicator. When the STA LED is glowing, the frequency is too high, go lower. When the CON LED is glowing, the frequency is too low, go higher.

Press the K key on your computer. Either the STA or CON LED should illuminate. Next, press the space bar on your keyboard and verify that the other LED (CON or STA) illuminates in place of the one illuminated in the preceding step. Press the space bar again until the CON LED is illuminated.

Slowly rotate the adjustment screw on trimpot R78 in the CCW direction until both the STA and CON LEDs illuminate. The LEDs are a bit slow to respond, so be careful when the STA LED begins to flicker. You will probably notice a portion of the adjustment when both LEDs illuminate; set the adjustment screw to the center of this band. If you go too far, only the STA LED will be illuminated; in this case, turn the adjustment screw CW.

When both LEDs are steadily illuminated, press the Q key. Both LEDs should extinguish and the command mode prompt should reappear. The lower tone is now calibrated.

To calibrate the upper tone, enter

cmd:**CALSET 239**

or use the value appropriate for the higher modulator tone. Next, enter

cmd:**CALIBRA**

Now press the K key. Either the CON or the STA LED should illuminate. Tap the space bar until the STA LED glows.

Slowly rotate the adjustment screw of trimpot R77 CW, following the instructions above for setting R78. Continue the adjustment until both the STA and CON LEDs glow steadily.

When both LEDs are steadily illuminated, press the Q key. Both LEDs should extinguish and the command mode prompt should reappear. Remove the jumpers at JMP9 pins 1 & 6 and at JMP4.

To calibrate the PLL demodulator, remove the jumper at JMP8. Next, preset R79 20 turns clockwise (CW) or until you hear the element."click" at full CW rotation. Enter

cmd:**CALSET 157**

or use the value appropriate for the demodulator frequency you are using. Next, enter

cmd:**CALIBRA**

Note that the STA and CON LEDs are both extinguished. Now place a jumper at JMP9 pins 2 and 5 (the center two pins). The CON LED should illuminate. If not, tap the space bar on your terminal until it does.

SLOWLY rotate the adjustment screw of R79 CCW until both the STA and CON LEDs glow. If these LEDs occasionally flicker while both are on, this is acceptable.

Remove the jumper at JMP9 and verify that the STA and CON LEDs extinguish.

Type Q and note that the command mode prompt reappears. Reinstall a jumper at JMP8. Your TNC 2 modem is now calibrated to the tones you have selected.

If you are operating at a data rate other than 1200 baud, the loop timing components on your demodulator will not be optimum; consult the Exar Application Note referenced in the Bibliography for further information. The table below lists the component values for the two most common FSK pairs as well as CALSET parameters.

Table 5-3. Modem Calibration Parameters.

Data rate	1200 baud	300 baud
Shift	1000 Hz	200 Hz
Center Freq	1685 Hz	1700 Hz
Upper Freq	2200 Hz	1800 Hz
Lower Freq	1200 Hz	1600 Hz
C54	2200 pF	0.01 uF
R93	47.5 k	226 k
CALSET Parameters		
High Tone (R77)	239	293
Low Tone (R78)	438	329
Demod PLL (R79)	157	155

NOTE: Components used for R93 and C54 should have temperature-stable characteristics.

Finally, it is best to match the IF bandwidth of your receiver to the data rate you will operate. For 300 baud HF operation, a filter of about 500 Hz bandwidth is optimum; any CW filter wider than this should be better than an SSB bandwidth filter!

Multi-Connect Survival Guide

This section is a very brief tutorial on the use of the multiple-connection capability included with this software release.

Why Multi-Connect?

Multiple connection capability is a very powerful addition to your TNC 2. It is very useful for traffic net operation, multi-user bulletin boards, path checking and so forth.

What is Multi-Connect?

Multiple connection operation is not the same as **multi-way** operation. With multiple connect, you may establish several point-to-point "links" with various stations. Multi-way, which is not available, would enable multiple stations to be simultaneously interconnected to each other, with each station seeing all data passed from any station in the group, error free.

Multiple connection operation is another step on the road to proper networking, and networking should eventually allow multi-way operation.

What Commands Set my TNC 2 to Normal Operation?

Your TNC 2 defaults the multi-connect-related commands to the following parameters:

```
CONPERM          OFF
STREAMCAll       OFF
STREAMDouBLe     OFF
STREAMSWitch     ¦
USERS            1
```

This sets up your TNC 2 to act just like a "normal" TNC does that doesn't support multiple connections. The key to obtaining this traditional operation is to set **USERS 1**.

How Do I Invoke Multi-Connect?

If **USERS** is not 1, you are telling the TNC to allow multiple connections to your stations from other stations. In addition, **TRANSPARENT** mode will operate differently, in that incoming data will be prefixed with the current **STREAMSW**itch character and identifier (such as "¦A"). Thus, truly transparent operation is not possible with this software release supporting multiple connections.

The **STREAMSW**itch character, defaulted to "¦". should be set to a character you won't normally use. Note that this character may be set to a hex value between $80 and $FF. This may allow you to use 8-bit characters (**AWLEN** 8) if your terminal or computer is capable of generating such "characters." This could help prevent

confusion in interpreting incoming data from other stations if they happen to send data that includes your selected **STREAMSW**itch character.

Although not foolfproof, enabling **STREAMDBL** may also help in sorting out **STREAMSW**itch characters included in the received data from a valid stream switches generated by your TNC 2.

STREAMCAll should be especially helpful when manually operating a station in which you allow multiple connections.

When in **CONVERSE** mode, you may switch streams by entering the **STREAMSW**itch character (default "¦"), followed by a stream identifier ("A" through "J"), followed by the data you wish to send to the station on that stream. See the example in the description of **STREAMCA**ll for an illustration of this.

If all this seems a bit confusing, don't worry, it is! The only way to really understand multiple connect operation is to try it! Tested and on-the-air, Once you have gained confidence in "normal" packet operation, try setting
USERS 2 and get a couple of friends to connect to you. Play with the commands (you can't hurt the TNC by issuing commands to it!) and see their effects.

MFJ TNC 2 COMMANDS

This chapter serves as a complete reference to all TNC commands. Commands are used to control the many variable values which affect your TNC's operation, as well as causing it to perform specific actions, such as connecting to another station to start a QSO. You can enter a command to your TNC whenever you see the command-mode prompt:

cmd:

You will change parameters and issue instructions to the TNC by typing commands composed of English-like words or word abbreviations, which are called keywords, and variables which are numbers or strings of characters chosen by the user. You will probably never change some of these parameters; however, one of TAPR's goals is to allow each user maximum flexibility to adapt the TNC to his environment.

Entering Commands

You may use either upper case (capital letters) or lower case (small letters) when you enter commands. In order to have your TNC accept a command line, you must finish the line with a <CR>, or carriage return character. This won't be mentioned explicitly in the examples below. Before you type the final <CR> of your command, you can correct typing mistakes or cancel the line completely. See Chapter 4 for a discussion of input line editing.

This chapter will use UPPER CASE for commands and lower case for explanatory text. In examples showing input typed by the user together with the responses of the TNC, the user's input will be shown in **bold face.**

Whenever the TNC accepts a command which changes a value, it will display the previous value. For example, if you type

XFLOW OFF

you might see the display

XFLOW was ON

This reminds you of what you have done, and indicates that the value has been successfully changed.

If you type something your TNC can't understand you will get an error message. If you type an unrecognized command, you will see the message **?EH**. If you get a command name correct, but the arguments are wrong, you will see the message **?BAD**. A complete list of error messages appears at the end of this chapter. For example,

```
cmd:ASDFASDF          ¦¦ This isn't a command.
?EH
cmd:BEACON E          ¦¦ A parameter was left out.
?BAD
```

Command List

The commands are listed alphabetically, and each command entry contains several sections, as follows.

COMMAND NAME

The boldface command name at the top of the entry is the word you will type in order to have your TNC execute this command. The command name is printed with some letters underlined. These letters form the minimum abbreviation that you may use and still have your TNC understand your command. Of course, you may type out the entire command word, or any abbreviation longer than the minimum abbreviation, if you like.

For example, the command **MYCALL** may be specified by simply typing **MY**. The abbreviation **M** is not sufficient (and will be interpreted as a different command), but **MY**, **MYC**, **MYCA**, **MYCAL** or **MYCALL** are all acceptable.

If the command requires parameters, they will be indicated after the command name.

Default

For commands that set values, your TNC assumes a "most often used" or default condition. The defaults are the values stored in EPROM which are loaded into RAM when the system is first powered up, or when you give the **RESET** command. Immediate commands perform actions rather than setting values, and don't have defaults.

Parameters

There are several types of parameters. Some parameters can have one of only two values, such as **ON** and **OFF** or **EVERY** and **AFTER**. If a parameter must be one of two values, the choices are shown separated by a vertical bar. You may use **YES** instead of **ON** and **NO** instead of **OFF**.

A parameter designated as **n** is a numeric value. These values may be entered as ordinary decimal numbers, or as hexadecimal, or "hex", numbers by preceding the number with a **$** symbol. When the TNC shows some of these numeric parameters (those which set special characters), they will be given in hex. The "digits" of a hex number represent powers of 16, analogous to the powers of 10 represented by a decimal number. The numbers 10 through 15 are denoted by the hex digits A through F. For example,

$1B = 1*16 + 11 = 27
$120 = 1*16*16 + 2*16 + 0 = 288

A parameter designated as **text**, such as the argument to **CTEXT**, may be entered in upper or lower case, and may include numbers, spaces, and punctuation. The text is accepted exactly as typed by the user.

Several commands require **callsign**s as parameters. While these parameters are normally Amateur call signs, they may actually be any collection of numbers and at least one letter up to six characters; they are used to identify stations sending and receiving packets. A **callsign** may additionally include sub-station ID (SSID), which is a decimal number from 0 to 15 used to distinguish two or more stations on the air with the same Amateur call (such as a base station and a repeater). The call sign and SSID are entered and displayed as call-n, e.g., KOPFX-3. If the SSID is not entered, it is set to 0, and SSIDs of 0 are not displayed by the TNC.

Some commands have parameters which are actually lists of items. For example, you may specify as many as eight callsigns to be selectively monitored with the command **LCALLS**. The second and later items in the list are optional, and you may separate the list items with blank spaces or with commas.

These examples may help you to understand the explanations above.

BEACON EVERY|AFTER n

means that the command **BEACON** requires an argument which must be either **EVERY** or **AFTER** (abbreviated to **E** or **A**), and an argument **n** which the user may choose from a range of values. An acceptable command might be **BEACON E 2**.

CONNECT call1 [VIA call2[,call3... ,call9]]

means that the command **CONNECT** requires a **callsign** argument **call1**. You may optionally include the keyword **VIA**, followed by a list of one to eight **callsigns**, **call2** through **call9**.

The **callsigns** in the list, if included, must be separated by commas (as shown), or by blank spaces. An acceptable command might be **C N2WX V AD7I WB9FLW.**

You can see the current value of the command's arguments by typing the command name by itself, without any arguments. For example,

```
cmd:CONOK Y          || Sets the value to YES (ON).
CONOK was OFF        || Displays previous value.
cmd:CONOK            || Command with no arguments
CONOK ON             ||   displays present value.
```

A special command, **DISPLAY**, allows you to see the values of all parameters or groups of related parameters.

Remarks

This section describes the command's action and the meaning of each argument. Examples may be included of situations in which the command might be used.

8BITCONV ON|OFF Default: OFF

Parameters:

ON	The high-order bit is not stripped in Converse Mode.
OFF	The high-order bit is stripped in Converse Mode.

This command enables transmission of 8-bit data in Converse Mode. If **8BITCONV** is **OFF**, the high-order bit (bit 7) of characters received from the terminal is removed before the characters are transmitted in a packet. The standard ASCII character set requires only 7 bits, and the final bit is used as a parity bit or ignored. Setting bit 7 in text characters transmitted over the air may cause confusion at the other end.

If you need to transmit 8-bit data, but don't want all the features of Transparent Mode, you should set **8BITCONV ON** and **AWLEN 8**. This may be desirable, for example, if you are using a special non-ASCII character set.

Bit 7 is always removed in Command Mode, since commands require only the standard 7-bit ASCII character set.

AUTOLF ON|OFF Default: ON

Parameters:

ON	A linefeed character (<LF>) is sent to the terminal after each carriage return character (<CR>).
OFF	A <LF> is not sent to the terminal after each <CR>.

AUTOLF controls the display of carriage return characters received in packets as well as echoing those that are typed in.

If the TNC's sign-on message lines appear to be typed over each other, you should set **AUTOLF ON**. If the TNC's sign-on message appears to be double-spaced, you should set **AUTOLF OFF**. If the TNC's sign-on message appears to be single-spaced, you have **AUTOLF** set correctly.

This command only affects what is displayed, not the data sent in packets. If you want to add linefeed characters to outgoing packets, use the command **LFADD**.

AWLEN n Default: 7

Parameters:

n	7 - 8, specifying the number of data bits per word.

This value defines the word length used by the serial IO terminal port.

For most packet operations, including conversation, bulletin board operation, and transmission of ASCII files, you should set **AWLEN 7**. If 8 bit words are transmitted to the TNC in Command Mode or Converse Mode, the eighth bit is normally removed, leaving a standard ASCII character, regardless of the setting of **AWLEN**.

To transmit and receive packets retaining all 8 data bits of each character, as you need to do if you send executable files or other special data, you should use Transparent Mode and set **AWLEN 8**. Alternatively, you can use Converse Mode and set **AWLEN 8** and **8BITCONV ON** (however, the data you then send must handle the Converse Mode special characters with the **PASS** prefix).

AX25L2V2 ON|OFF Default: OFF

Parameters:

ON	The TNC will use AX.25 Level 2 Version 2.0 protocol.
OFF	The TNC will use AX.25 Level 2 Version 1.0 protocol.

Some implementations of the earlier version of AX.25 protocol (e.g., TAPR's TNC 1) won't properly digipeat version 2.0 AX.25 packets. This command exists to provide compatibility with these other TNCs until their software has been updated.

During the protocol transition period, you should set **AX25L2V2 OFF**.

After your local area TNCs are updated to the newer protocol version, you should set **AX25L2V2 ON**.

AXDELAY n Default: 0

Parameters:

n	0 - 180, specifying the voice repeater keyup delay in 10 ms intervals.

AXDELAY specifies a period of time the TNC is to wait, in addition to the normal delay set by **TXDELAY**, after keying the transmitter and before data is sent. This feature will be used by groups using a standard "voice" repeater to extend the range of the local area network. Repeaters with slow mechanical relays, split sites, or other circuits which delay transmission for some time after the RF carrier is present require some amount of time to get RF on the air.

If you are using a repeater that hasn't been used for packet operations before, you will have to experiment to find the best value for **n**. If other packet stations have been using the repeater, check with them for the proper setting. Note that this command acts in conjunction with **AXHANG**.

Note that the TAPR TNC 1 and other TNCs using the same version 3.x firmware interpret **n** in 120 ms intervals. The value set by **AXDELAY** on TNC 2 will thus be 12 times the value used by a TNC 1 user to give the same delay time.

AXHANG n Default: 0

Parameters:

n 0 - 20, specifying the voice repeater hang time in 100 ms intervals.

This value can be used to increase channel efficiency when an audio repeater with a hang time greater than 100 ms is used. For a repeater with a long hang time, it is not necessary to wait for the repeater keyup delay after keying the transmitter if the repeater is still transmitting. If the TNC has heard a packet sent within the hang period, it will not add the repeater keyup delay (**AXDELAY**) to the keyup time.

If you are using a repeater that hasn't been used for packet operations before, you will have to experiment to find the best value for **n**. If other packet stations have been using the repeater, check with them for the proper setting.

Note that the TAPR TNC 1 and other TNCs using the same version 3.x firmware interpret **n** in 120 ms intervals. The value you set on TNC 2 for **AXHANG** will thus be 6/5 the value used by a TNC 1 user for the same hang time (when converting, round down to the nearest integer).

BEACON EVERY|AFTER n Default: EVERY 0

Parameters:

EVERY	Send beacon at regular intervals.
AFTER	Send beacon once after the specified time interval with no packet activity.
n	0 - 250, specifying beacon timing in 10 second intervals. A value of 0 disables the beacon.

This command enables beacon sending and causes the first beacon frame to be transmitted. A beacon frame consists of the text specified by **BTEXT** in a packet addressed to "BEACON" and sent via the digipeat addresses specified by the **UNPROTO** command, if any.

If the keyword **EVERY** is specified, a beacon packet is sent every **n***10 seconds. This mode might be used to transmit packets for testing purposes.

If **AFTER** is specified, a beacon is sent only after **n***10 seconds have passed with no packet activity. In this case, the beacon is sent only once until further activity is detected. This mode can be used to send announcements or test messages only when packet stations are on the air. If you choose **n** properly you can avoid cluttering a busy channel with unnecessary transmissions.

Beacon frames from other TNCs can be monitored by setting **MONITOR ON**.

BKONDEL ON|OFF Default: ON

Parameters:

ON — The sequence <BACKSPACE> <SPACE> <BACKSPACE> is echoed when a character is deleted from the input line.

OFF — The <BACKSLASH> character (\) is echoed when a character is deleted.

This command determines the way the display is updated to reflect a character deletion in Command Mode or Converse Mode.

The <BACKSPACE> <SPACE> <BACKSPACE> sequence will properly update the screen of a video display. If you have a video display terminal or computer, you should set **BKONDEL ON**.

The <BACKSPACE> <SPACE> <BACKSPACE> sequence on a printing terminal would result in overtyped text. If you have a paper-output display, or if your terminal does not respond to the <BACKSPACE> character (<CTRL-H>), you should set **BKONDEL OFF**. The TNC will type a <BACKSLASH> for each character you delete. You can display the corrected input line by typing the redisplay-line character, which is set by the command **REDISPLA**.

BTEXT text Default: ""

Parameters:

text Any combination of characters and spaces up to a maximum length of 120 characters.

BTEXT specifies the content of the data portion of a beacon packet. The default **text** is an empty string, i.e., no message. Beacon packets are discussed in more detail under the **BEACON** command.

You can send multiple-line messages in your beacon by including carriage return (<CR>) characters in the text. The <CR> character can be included by preceding it with the pass character. The pass character is set by the **PASS** command. If you enter a text string longer than 120 characters, an error message will appear and the command will be ignored.

For example, a Bulletin Board program might set the beacon **text** to a message like this, updating the **text** after each connection:

Mailbox on line. Messages for WB9FLW, AD7I, K9NG.

To clear the **BTEXT text** without issuing the **RESET** command, use a % or & character as the first character in the text.

BUDLIST ON|OFF Default: OFF

Parameters:

ON — Ignore frames from stations which are not in the **LCALLS** list.

OFF — Ignore frame from stations which are in the **LCALLS** list.

BUDLIST works in conjunction with the command **LCALLS**, which sets up a **callsign** list. These commands determine which packets will be displayed when you have set **MONITOR ON**. **BUDLIST** specifies whether the **callsigns** in the list are the ones you want to ignore or, alternatively, are the only ones you want to listen to.

If you want to listen only for packets from a limited list, you should enter this list with **LCALLS** and set **BUDLIST ON**. You can use this feature, for example, to have your TNC "keep an ear out" for a particular station while you converse with someone else.

If you want to ignore packets from a limited list, you should list the **callsigns** to ignore in **LCALLS** and set **BUDLIST OFF**. For example, if there is a bulletin board on frequency, you can ignore it while monitoring other conversations.

CALIBRA

CALIBRA is an immediate command, and is used to transfer control to the modem calibration routine. Calibration may be performed at any time without altering the current link state.

Briefly, the commands available in the calibration routine are:

<SPACE> Switch the transmit audio to the other tone.
D Alternate between the two transmit tones at a rate determined by the radio baud rate.
K Toggle PTT line on/off.
Q Quit calibration routine.

Calibration of the modem tones is described in the modem calibration section of Chapter 5. Adjustment of the tone levels using **CALIBRA** is described in Chapter 3.

CALSET n Default: NONE

Parameters:

n 0 - 65535, specifying the count setting for use by the calibration routine.

The number to be specified by **CALSET** is determined by the frequency **f** to be calibrated as follows.

n = (525,000 / **f**) + 1 (modulator tones)
n = (262,500 / **f**) + 1 (demodulator tones)

Round **n** to the nearest integer. To calibrate the modem tones, you will set **n** for the desired frequency using **CALSET**, enter the calibration routine with **CALIBRA**, and adjust the tone following the instructions in Chapter 5. To calibrate the next tone, exit the calibration routine and set **n** for the next frequency.

CANLINE n Default: $18 <CTRL-X>

Parameters:

n 0 - $7F, specifying an ASCII character code.

This command is used to change the cancel-line input editing command character. The parameter **n** is the ASCII code for the character you want to type in order to cancel an input line. You can enter the code in either hex or decimal.

For example, to change the cancel-line character to <CTRL-U>, you would set **CANLINE $15** or **CANLINE 21**.

If you cancel an input line in Command Mode, the line will be terminated with a <BACKSLASH> character, and you will see a new **cmd:** prompt. If you cancel a line in Converse Mode, you will see only the <BACKSLASH> and a new line. You can cancel only the line you are currently typing. Once a <CR> has been entered, you can not cancel an input line using the cancel-line character. Note that if your send-packet character is not <CR>, the cancel-line character will cancel only the last line of a multi-line packet. To cancel the entire packet, use the **CANPAC** character.

Line cancellation, like all other input editing features, is disabled in Transparent Mode.

CANPAC n Default: $19 <CTRL-Y>

Parameters:

n 0 - $7F, specifying an ASCII character code.

CANPAC is used to change the cancel-packet input editing command character. The parameter **n** is the ASCII code for the character you want to type in order to cancel an input packet. You can enter the code in either hex or decimal.

If you cancel a packet in Converse Mode, the line will be terminated with a <BACKSLASH> character and a new line. You can only cancel the packet that is currently being entered. Once you have typed the send-packet character, or waited **PACTIME** (if **CPACTIME** enabled), the packet can not be cancelled even if it has not been transmitted.

Packet cancellation, like other input editing features, is disabled in Transparent Mode.

Cancel Display Output

The cancel-packet character also functions to cancel display output in Command Mode. If you are in Command Mode and type the cancel-packet character, any characters that would be typed on the screen (except echoed characters) are "thrown away" by the TNC. Typing the cancel-output character a second time restores normal output. To see how this works, try typing **DISPLAY**, then type a <CTRL-Y>. The command list display will stop. You won't see any response from the TNC to commands. Now type another <CTRL-Y>, and type **DISPLAY** again to see that the display is back to normal.

You can use the cancel-display feature if you inadvertently do something that causes the TNC to generate large amounts of output to the terminal, such as giving the **DISPLAY** command or setting **TRACE ON**. If you are in Converse Mode or Transparent Mode and want to cancel display output, you must exit to Command Mode and then type the cancel-packet character.

CHECK n Default: 30

Parameters:

n 0 - 250, specifying the check time in 10 second intervals. A value of 0 disables this feature.

This command sets a connection timeout. If a link connection exists between your TNC and another station, and the other station "disappears," your TNC could remain in the connected state indefinitely, refusing connections from other stations. This might happen if propagation changes unexpectedly or an intermediate digipeater station is turned off. In order to prevent this sort of lockup, the TNC will try to clean up the link if the specified time elapses without any packets being heard from the other TNC. The operation of this feature depends on the setting of **AX25L2V2**.

If **AX25L2V2** is **ON**, the TNC will send a "check packet" to verify the presence of the other station if no packets have been heard from it for **n***10 seconds. This frame contains no information, but is interpreted by the receiving station's TNC as an inquiry as to whether it is still connected. If the receiving TNC is still connected, it sends an appropriate response packet. If the TNC initiating the inquiry does not hear a response after RETRY+1 attempts, it commences a disconnect sequence, as if the **DISCONNE** command had been given.

If **AX25L2V2** is **OFF** and the other station has not been heard for **n***10 seconds, the TNC will not attempt an inquiry, but will send a disconnect packet, just as if you had typed the command **DISCONNE**.

CMDTIME n Default: 1

Parameters:

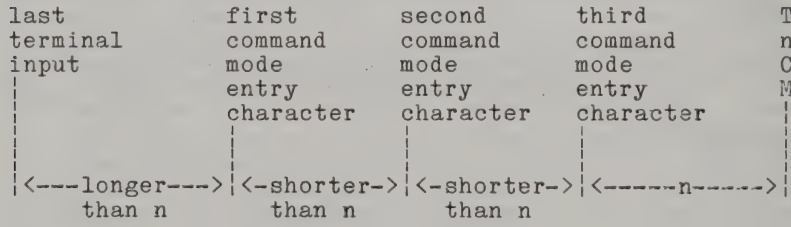
n 0 - 250, specifying Transparent Mode timeout value in 1 second intervals. If **n** is zero, the only exit from Transparent Mode is to send a BREAK signal or interrupt power to the TNC.

This command sets the Transparent Mode timeout value. In order to allow escape to Command Mode from Transparent Mode while permitting any character to be sent as data, a guard time of **n** seconds is set up.

The same Command Mode entry character used for exit from Converse Mode is used to exit Transparent Mode, but the procedure is different. (The Command Mode entry character is set by **COMMAND**.) Three Command Mode entry characters must be entered less than **n** seconds apart, with no intervening characters, after a delay of **n** seconds since the last characters were typed. After a final delay of **n** seconds, the TNC will exit Transparent Mode and enter Command Mode. You should then see the prompt

cmd:

The diagram below illustrates this timing.



CMSG ON|OFF Default: OFF

Parameters:

ON	A text message is sent as the first packet after a connection is established.
OFF	The text message is not sent.

CMSG enables automatic sending of the message set by **CTEXT** whenever your TNC accepts a connect request from another TNC.

For example, if you have left your station running even though you don't want to operate just now, you might want to set **CMSG ON** to let people know that you can't talk when they connect to your TNC. When you are ready to operate, you would set **CMSG OFF**.

COMMAND n Default: $03 <CTRL-C>

Parameters:

n	0 - $7F, specifying an ASCII character code.

This command is used to change the Command Mode entry character. You can enter the code in either hex or decimal.

Command Mode is entered from Converse Mode when this character is typed. If you type the Command Mode entry character while you are already in Command Mode, nothing will happen. To see how this works, enter Converse Mode by typing **CONVERS**. Anything you type will become packet data. Now type a <CTRL-C>. You will see the Command Mode prompt, indicating successful exit to Command Mode. The display might look like this:

```
     cmd:CONVERS
     Hello World!  I'm on the air on packet radio!
[enter <CTRL-C>]
     cmd:
```

See the entry under **CMDTIME** or the discussion of Transparent Mode in Chapter 5 for information on how the Command Mode entry character is used for escape from Transparent Mode.

CONMODE CONVERS¦TRANS Default: CONVERS

Parameters:

CONVERS Sets automatic entry to Converse Mode when a connection is established.

TRANS Sets automatic entry to Transparent Mode when a connection is established.

CONMODE controls which mode the TNC will be placed in after a connection. The connection may result either from a connect request received over the air or a connect initiated by a **CONNECT** command that you issue. For most operations, you would set **CONMODE** to **CONVERS**. However, if you are using Transparent Mode for a bulletin board program, for example, you would set **CONMODE** to **TRANS** so that the correct mode will be entered when your bulletin board receives a connect request.

If you initiate a connection with the **CONNECT** command, the timing of the entry into Converse or Transparent Mode is determined by **NEWMODE**.

If the TNC is already in Converse or Transparent Mode when the connection is completed, the mode will not be changed. If you have typed part of a command line when the connection is completed, the mode change will not take place until you complete the command or cancel the line. This prevents the last part of your command from inadvertently being sent as a packet.

CONNECT call1 [VIA call2[,call3... ,call9]]

Parameters:

call1 **Callsign** of TNC to be connected to.

call2 Optional **callsign** of TNC to be digipeated through. As many as eight digipeat addresses can be specified.

The part of the command line in brackets, **VIA call2[, call3... ,call9]** is optional. The double-bracketed text, **,call3... ,call9**, is also optional, but would only be used if **VIA call2** is present. The brackets are not typed.

Each **callsign** may include an optional sub-station ID specified as **-n** immediately following the **callsign**. The digipeat fields are specified in the order in which you want them to relay the packets to the destination, **call1**.

CONNECT is an immediate command. It initiates a connect request to TNC **call1**, optionally through digipeaters. If **NEWMODE** is **ON**, the TNC will immediately enter Converse Mode or Transparent Mode, as specified by the command **CONMODE**. If **NEWMODE** is **OFF**, the TNC will enter Converse Mode or Transparent Mode when the connection is successfully completed.

An error message is returned if the TNC is in a connected state, or is already attempting to connect or disconnect. If no response to the connect request occurs after the number of attempts specified by **RETRY**, the command is aborted and a message is typed. The TNC returns to Command Mode if **NEWMODE** is **ON**. If **NEWMODE** is **OFF**, the mode does not change, i.e. the TNC remains in Command Mode.

For example, to connect to WA7GXD using NOADI-1 (who is near your QTH) and WDOETZ (who is near GXD's QTH) as digipeaters, you would type

CONNECT WA7GXD VIA NOADI-1 ,WDOETZ

Packets coming back from WA7GXD access the digipeaters in the opposite order. Thus, packets from WA7GXD will first be repeated by WDOETZ, then by NOADI-1.

CONOK ON|OFF Default: ON

Parameters:

ON Connect requests from other TNCs will be accepted.

OFF Connect requests from other TNCs will be rejected.

This command determines the action taken by the TNC when a connect request for it is received though the radio. If **CONOK** is **ON**, the request will be acknowledged, the standard connect message will be typed and either Converse or Transparent Mode will be entered, depending on the setting of **CONMODE**.

If **CONOK** is **OFF** and the TNC is not in Transparent Mode, the TNC will notify you of the connect attempt with the following message.

connect request: <call>

The **callsign** of the station trying to connect will replace **<call>**.

All connect requests from stations with totally blank callsign are rejected with a busy response.

The TNC will also issue a DM packet, or "busy signal" to the requesting station. The user may then issue his own connect command. If your TNC receives a DM packet in response to a connect request, it will type the message

*** <call> station busy

with the **callsign** of the station that sent the DM packet in place of **<call>**.

For example, if you want to leave your station running as a digipeater you might set **CONOK OFF** until you are ready for a conversation. If you get a connect request in the meantime, you can change your mind. Stations attempting to connect to your TNC will be able to see that the station is up but not available for connection (it might still be useful as a digipeater).

CONSTAMP ON|OFF Default: OFF

Parameters:

ON Connect and disconnect status messages are time stamped.

OFF Connect status messages are not time stamped.

This command enables time stamping of ***** CONNECTED** status messages. The date and time information is then available for use by Bulletin Board programs or other computer applications. The date and time must be set initially by the **DAYTIME** command before time stamping will occur. The date format is determined by the **DAYUSA** command.

For example, if **CONSTAMP** is **ON**, **DAYUSA** is **ON**, and the date and time have been set, a connect message might appear as follows.

*** CONNECTED to N2WX [05/28/85 16:28:31]

CONVERS

CONVERS is an immediate command, and will cause the TNC to exit from Command Mode into Converse Mode. Any link connections are not affected. Once in Converse Mode, everything you type is packetized and transmitted over the radio. Typing the Command Mode entry character returns the TNC to Command Mode. See the discussions of Converse Mode in Chapter 4 and Chapter 5.

CPACTIME ON|OFF Default: OFF

Parameters:

ON	Packet timeout is used in Converse Mode.
OFF	Packet timeout is not used in Converse Mode.

This command enables the periodic automatic sending of packets in Converse Mode. This feature may be used for computer communications, such as Bulletin Board operation, when the full Transparent Mode features are not desired.

If **CPACTIME** is **ON**, characters are packetized and transmitted periodically as they are in Transparent Mode, but local editing and display features of Converse Mode are enabled, and software flow control may be used. For a discussion of how periodic packetizing works, see the command **PACTIME**, which controls the rate and mode of packet assembly.

You should set **CR OFF** in this mode, since otherwise the send-packet character will be inserted in the data being packetized even though it was not typed. In order to include <CR> characters in transmitted packets, set **SENDPAC** to a normally unused character (e.g., <CTRL-P>), at which point the TNC will treat <CR> as an ordinary character.

You can set **CPACTIME ON** for a mode of operation similar to full break-in CW, in which your text is transmitted soon after you type it, but in short bursts of a few characters, and the other station may break in at will. Some operators find it easier to carry on a conversation in this mode, since it eliminates the delays while long packets are being typed.

CR ON|OFF Default: ON

Parameters:

ON — The send-packet character, normally <CR>, is appended to all packets sent in Converse Mode.

OFF — The send-packet character is not appended to packets.

When **CR** is **ON**, all packets sent in Converse Mode will include, as the last character of the packet, the send-packet character which forces the packet to be sent. If **CR** is **OFF**, the send-packet character is interpreted solely as a command to the TNC, not as data to be included in the packet, and furthermore, it will not be echoed to the terminal.

Setting **CR ON** and **SENDPAC $OD** results in a natural conversation mode. Each line is sent when a <CR> is entered, and arrives at its destination with a <CR> at the end of the line. If the station at the other end reports overprinting of lines on his display, you can set **LFADD ON**, or the other station can set **AUTOLF ON**.

CTEXT text Default: ""

Parameters:

text — Any combination of characters and spaces, up to a maximum length of 120 characters.

CTEXT specifies the text of the packet to be sent after a connection is made, if **CMSG** is **ON**. The default **text** is an empty string, i.e., no message.

You can send multiple-line messages by including carriage return (<CR>) characters in the text. The <CR> character can be included by using the pass character immediately preceding it (see the **PASS** command). If you enter a **text** string longer than 120 characters, an error message will appear and the command will be ignored.

For example, you might set your **CTEXT** message to

I'm not here right now, but you may leave a message.

To clear the **CTEXT text** without issuing a **RESET** command, use a % or & as the first character in the message.

You cannot connect to yourself and see your CTEXT. CMSG only takes effect if you are connected to by another TNC.

DAYTIME date&time No default

Parameters:

date&time Current date and time to set.

This command allows you to set the current date and time for the TNC. The format for entering the **date&time** is

yymmddhhmm

where **yy** is the last two digits of the year, **mm** is the two-digit month code (01-12), **dd** is date (01-31), **hh** is the hour (00-23), and **mm** is the minutes after the hour (00-59). All these codes must be exactly two digits, so that numbers from 0 to 9 must be entered with leading zeros. The TNC does not check thoroughly for the correct number of days in a month, so you should use some judgment when you set the date.

The **date&time** parameter is used by the commands **CONSTAMP** and **MSTAMP** to "time stamp" received and monitored messages. Entries in the "heard" (displayed by **MHEARD**) are also time stamped if **date&time** has been set. The TNC's time is updated continuously as long as it is powered up. You must reset the date and time each time you turn on the TNC. If you don't do this, the commands **CONSTAMP** and **MSTAMP** will not enable time stamping.

If you type **DAYTIME** without a parameter, the TNC will display the current date and time. The format of the display is **dd-mm-yy hh:mm** if **DAYUSA** is **OFF**, and **mm/dd/yy hh:mm** if **DAYUSA** is **ON**. The format for entering **date&time** is not affected. For example,

cmd:**DAYTIME 8402291530**

sets the date and time to February 29, 1984 at 3:30 PM. The display of the date and time, with **DAYUSA ON** would be:

cmd:**DAYTIME**
02/29/84 15:30:26

DAYUSA ON¦OFF Default: ON

Parameters:

ON Date is displayed in the format **mm/dd/yy.**

OFF Date is displayed in the format **dd-mm-yy.**

This command determines the format for the TNC's display of the date. If **DAYUSA** is **ON**, the standard U.S. format is used; if **DAYUSA** is **OFF**, the standard European format is used. This command affects the format of the date display used in "time stamps" as well as the display when **DAYTIME** is entered without parameters. The format for entering the time using **DAYTIME** is not affected.

For example, if **DAYUSA** is **ON**, then July 2, 1984 at 9:28:44 AM would be displayed as

```
cmd:DAYTIME
07/02/84 9:28:44
```

If **DAYUSA** is **OFF** the same date and time would appear as

```
cmd:DAYTIME
02-07-84 9:28:44
```

DELETE ON¦OFF Default: OFF

Parameters:

ON The delete character input editing character is <DELETE> ($7F).

OFF The delete character input editing character is <BACKSPACE> ($08).

This command is used to change the input editing command for character deletion. When this character is typed, the last character from the input line is deleted. How the TNC indicates the deletion is controlled by the **BKONDEL** command.

You can not use this character to delete past the beginning of a line, although you can delete <CR> characters that have been entered in the text with the pass character.

To see a corrected display of the current line after you have deleted characters, type the redisplay-line character, which is set by the **REDISPLA** command.

DIGIPEAT ON|OFF Default: ON

Parameters:

ON The TNC will digipeat packets if requested.

OFF The TNC will not digipeat packets.

When this parameter is **ON**, any packet received that has your TNC's **callsign** (including SSID) in the digipeat list of its address field will be retransmitted. Each station included in the digipeat list relays the packet in its turn, marking the packet so that it will not accidentally relay it twice (unless so requested), and so that the stations will relay the packet in the correct order. Digipeating takes place concurrently with other TNC operations and does not interfere with normal operation of a packet station.

In the spirit of cooperation typical of Amateur operation, you will probably want to set **DIGIPEAT ON** most of the time. However, you might want to disable digipeating if you're not home, or if your transmit relay makes enough noise to wake you up at night.

The command **HID** enables automatic transmission of identification packets if your station is acting as a digipeater.

DISCONNE

DISCONNE is an immediate command. It will initiate a disconnect request with the currently connected station. A successful disconnect results in the display of:

*** DISCONNECTED

Other commands may be entered while the disconnect is taking place, although connects are disallowed until the disconnect is completed. If the retry count is exceeded while waiting for the other side to acknowledge, the TNC moves to the disconnected state. If a disconnect command is entered while the TNC is disconnecting, the retry count is immediately set to the maximum number. In either case, the disconnect message is

*** retry count exceeded
*** DISCONNECTED

Disconnect messages are not displayed when the TNC is in Transparent Mode.

DISPLAY [class]

Parameters:

class — Optional parameter-class identifier, one of the following:

ASYNC	display asynchronous port parameters
CHARACTE	display special characters
ID	display ID parameters
LINK	display link parameters
MONITOR	display monitor parameters
TIMING	display timing parameters

DISPLAY is an immediate command, and with no **class** parameter will cause all control parameters and their current values to be displayed. Sub-groups of related parameters can be displayed by specifying the optional parameter-class. Individual parameters can be displayed by entering the parameter name with no options.

DWAIT n Default: 16

Parameters:

n — 0 - 250, specifying default wait time in 10 ms intervals.

This value is used to avoid collisions with digipeated packets. The TNC will wait the default wait time after last hearing data on the channel before it begins its own keyup sequence, unless the TNC is waiting to transmit digipeated packets. This value should be agreed on by all members of a local area when digipeaters are used in the area. The best value will be determined by experimentation, but will be a function of the keyup time (**TXDELAY**) of the digipeater stations.

This feature is intended to help alleviate the drastic reduction of throughput that occurs on a channel when digipeated packets suffer collisions. It is necessary because digipeated packets are not retried by the digipeater, but must be restarted by the originating station. If all stations specify a default wait time, and the right value of **n** is chosen, the digipeater will capture the frequency every time it has data to send, since digipeated packets are sent without this delay.

ECHO ON¦OFF Default: ON

Parameters:

ON — Characters received from the computer or terminal are echoed by the TNC.

OFF — Characters are not echoed.

This command controls local echoing by the TNC when it is in Command or Converse Mode. Local echoing is disabled in Transparent Mode.

If you don't see your input on the display, you should set **ECHO ON**. If you see two copies of every character you type, you should set **ECHO OFF**. If you see the characters you type displayed correctly, you have **ECHO** set correctly.

ESCAPE ON¦OFF Default: OFF

Parameters:

ON — The <ESCAPE> character ($1B) is output as "$" ($24).

OFF — The <ESCAPE> character is output as <ESCAPE> ($1B).

This command specifies the character which will be output when an <ESCAPE> character is to be sent to the terminal. The <ESCAPE> translation is disabled in Transparent Mode.

This command is provided because some terminals, and computer programs that emulate such terminals, interpret the <ESCAPE> character as a special command prefix. Such terminals may alter their displays depending on the characters following the <ESCAPE>. If you have such a terminal, you can protect yourself from unexpected text sequences and from other packeteers by setting **ESCAPE ON**.

See also the **MFILTER** command, which allows general character stripping (rather than character translation) in monitored packets.

FLOW ON|OFF Default: ON

Parameters:

ON	Type-in flow control is active.
OFF	Type-in flow control is disabled.

When type-in flow control is enabled, any character entered from the terminal will halt output to the terminal until: (1) a packet is forced (in Converse Mode); (2) a line is completed (in Command Mode); (3) the packet length is exceeded; or (3) the terminal output buffer fills up. Cancelling the current command or packet or typing the redisplay-line character will also cause output to resume. Type-in flow control is not used in Transparent Mode.

Setting **FLOW ON** will keep received data from interfering with data entry. If you (and the person you are talking to) wait for a packet from the other end before starting to respond, you can set **FLOW OFF**. Some Bulletin Board programs may work best with **FLOW OFF**. Some computers with "software UARTs" may be unable to send and receive data at the same time; users of such computers should set **FLOW ON**.

FRACK n Default: 3

Parameters:

n	1 - 15, specifying frame acknowledgment time-out in 1 second intervals.

After transmitting a packet requiring acknowledgment, the TNC will wait for the frame acknowledgment timeout before incrementing the retry counter and sending the frame again. If the packet address includes relay requests, the time between retries will be adjusted to

Retry interval = **n** * (2*m + 1)

where m is the number of intermediate relay stations.

When a retried packet is sent, a random wait time is added to any other wait times in use. This is to avoid lockups in which two TNCs repeatedly send packets which collide with each other.

FULLDUP ON|OFF Default: OFF

Parameters:

ON Full duplex mode is enabled.

OFF Full duplex mode is disabled.

When full duplex mode is disabled, the TNC makes use of the Data Carrier Detect signal from the modem to avoid collisions, and acknowledges multiple packets in a single transmission with a single acknowledgment. When full duplex mode is enabled, the TNC ignores the DCD signal and acknowledges packets individually. The latter mode is useful for full-duplex radio operation, such as through OSCAR 10. It should not be used unless both your station and the station you are communicating with are full-duplex stations.

You may find full-duplex mode useful for some testing operations, such as analog- or digital-loopback tests.

HEADERLN ON|OFF Default: OFF

Parameters:

ON The header for a monitored packet is printed on a separate line from the packet text.

OFF The header and packet text of monitored packets are printed on the same line.

This command affects the display format for monitored packets. If **HEADERLN** is **OFF**, the address information is displayed with the packet:

```
KV7D>N2WX: Go ahead and transfer the file.
```

If **HEADERLN** is **ON**, the address information is displayed, followed by the packet text on a separate line:

```
N2WX>KV7D:
Sorry, I'm not quite ready yet.
```

If you have set **MRPT ON** or enabled **MSTAMP**, you may wish to set **HEADERLN ON**, as the packet header quickly becomes long enough to fill a screen when these functions are active.

HID ON¦OFF Default: OFF

Parameters:

ON	Enables HDLC identification by a digipeater.
OFF	Disables HDLC identification.

This command is used to enable or disable the sending of identification packets by the TNC. If **HID** is **OFF**, the TNC will never send an identification packet. If **HID** is **ON**, the TNC will send an identification packet every 9.5 minutes if the station is digipeating packets. The **ID** command allows the operator to send a final identification packet if the station is being taken off the air.

An identification consists of an unsequenced I frame whose data field is your station identification. The identification packet is addressed to the "CQ" address set by the **UNPROTO** command. Your station identification is your callsign as set by **MYCALL**, with "/R" appended.

ID

ID is an immediate command. It will send a special identification packet. **ID** can be used to force a final identification packet to be sent as a digipeater station is being taken off the air. The identification packet will be sent only if the digipeater has transmitted since the last automatic identification.

An identification consists of an unsequenced I frame whose data field is your station identification. The identification packet is addressed to the "CQ" address set by the **UNPROTO** command. Your station identification is your callsign as set by **MYCALL**, with "/R" appended.

LCALLS call1[,call2... ,call8] Default: ""

Parameters:

call **Callsign** list. Up to 8 calls, separated by commas.

Each **callsign** may include an optional sub-station ID specified as **-n** immediately following the call. This command works in conjunction with **BUDLIST** and allows selective monitoring of other packet stations. These two commands determine which packets will be displayed when you have set **MONITOR ON**. **BUDLIST** specifies whether the **callsigns** in the list are the ones you want to ignore or, alternatively, are the only ones you want to listen to.

If you want to listen only for packets from a limited list, you should enter your selected list with **LCALLS** and set **BUDLIST ON**.

If you want to ignore packets from a limited list, you should list the **callsigns** to ignore in **LCALLS** and set BUDLIST OFF.

"%" and "&" may now be used to clear the LCALLS list.

LCOK ON|OFF **Default: ON**

Parameters:

ON	The TNC will send lower case characters to the computer or terminal.
OFF	The TNC will translate lower case characters to upper case.

If LCOK is OFF, lower case characters will be translated to upper case before being output to the terminal. This case translation is disabled in Transparent Mode. Input characters and echoes are not case translated.

If your computer or terminal does not accept lower case characters it may react badly if the TNC sends such characters to it. This command allows you to translate all lower case characters received in packets, as well as messages from the TNC, to upper case.

Since echoes of the characters you type are not translated to upper case, you can use this command to make your display easier to read when you are conversing in connected mode. If you and the other station's operator set LCOK OFF, you can each type your own messages in lower case and see incoming packets displayed in upper case. It will then be easy to distinguish incoming and outgoing lines.

LFADD ON|OFF **Default: OFF**

Parameters:

ON	A <LF> character is added to outgoing packets following each <CR> transmitted in the packet.
OFF	No <LF> is added to outgoing packets.

This function is similar to AUTOLF, except that the <LF> characters are added to outgoing packets rather than to text displayed locally. This feature is included in order to maintain compatibility with other packet radio controllers. If the person you are talking to reports overprinting of packets from your station you should set LFADD ON. This character insertion is disabled in Transparent Mode.

MALL ON|OFF Default: ON

Parameters:

ON Monitored packets include both "connected" packets and "unconnected" packets.

OFF Monitored packets include only "unconnected" packets.

This command determines the class of packets which are monitored. If MALL is OFF, only otherwise eligible packets (as determined by the BUDLIST and LCALLS commands) sent by other TNCs in the unconnected mode are displayed. This is the normal manner of operation when this TNC is being used to talk to a group of TNCs all of which are unconnected.

If MALL is ON, all otherwise eligible frames are displayed, including those sent between two other connected TNCs. This mode may be enabled for diagnostic purposes or for "reading the mail."

MAXFRAME n Default: 4

Parameters:

n 1 - 7, signifying a number of packets.

MAXFRAME sets an upper limit on the number of unacknowledged packets which the TNC can have outstanding at any one time. This is also the maximum number of contiguous packets which can be sent during any given transmission. If some but not all of the outstanding packets are acknowledged, a smaller number may be transmitted the next time, or new frames may be included in the retransmission, so that the total unacknowledged does not exceed n.

If you perform file transfers, you should experiment with MAXFRAME and PACLEN. If the link is good, there is an optimum relationship between the parameters set by these commands so that the maximum number of characters outstanding does not exceed the packet receive buffer space of the TNC receiving the data.

MCOM ON¦OFF Default: OFF

Parameters:

ON Connect, disconnect, UA, and DM frames are monitored.

OFF Only information frames are monitored.

This command enables monitoring of connect and disconnect frames when MONITOR is ON.

When MCOM is OFF, only I frames (packets containing user information) will be displayed. When MCOM is ON four protocol packets will also be displayed. Connect, disconnect, UA, and DM packets that are monitored are indicated by <C>, <D>, <UA>, and <DM>, respectively. As with other monitor commands, the stations monitored are determined by BUDLIST and LCALLS.

MCON ON¦OFF Default: ON

Parameters:

ON Monitor mode remains active when TNC is connected.

OFF Monitor mode is off while the TNC is connected.

If MCON is ON, the MONITOR command will enable monitoring while your TNC is connected to another TNC. If MCON is OFF, the display of monitored packets is suspended when a connect occurs, and is resumed when the TNC is disconnected.

If you want to see all packets displayed when you are not connected but have such display suppressed when you connect to another station, you should set MCON OFF.

MFILTER n1[,n2[,n3[,n4]]] **Default: none**

Parameters:

n	0 - $7F, specifying an ASCII character code. Up to 4 characters may be specified.

This command allows you to specify characters to be "filtered," or eliminated from monitored packets. The parameters n1, n2, etc., are the ASCII codes for the characters you want to filter. You can enter the code in either hex or decimal.

For example, if a <CTRL-L> character causes your screen to be cleared, and you don't want this to happen, you can set MFILTER 12. If you also want to eliminate <CTRL-Z> characters, which some computers interpret as end-of-file markers, you can set MFILTER 12,26.

MHCLEAR

MHCLEAR is an immediate command. It causes the list of stations heard to be cleared. You can use this command in conjunction with MHEARD to keep track of the stations on the air over a given period of time, such as an evening or a week. Clear the list of stations heard when you first begin to monitor the packet activity.

MHEARD

MHEARD is an immediate command. It causes the TNC to display the list of stations that have been heard since the last time the command MHCLEAR was given. Stations that are heard through digipeaters are marked with a * in the heard log. If you clear the list of stations heard at the beginning of a session, you can use this command to easily keep track of the stations that are active during that period. The maximum number of heard stations that can be logged is 18. If more stations are heard, earlier entries are discarded. Logging of stations heard is disabled when PASSALL is ON.

If the DAYTIME command has been used to set the date and time, entries in the heard log will be time stamped. For example,

```
cmd:MHEARD
K4NTA*     06/09/85  21:08:19
WA4ITD*    06/09/85  21:08:17
W1BEL-2*   06/09/85  21:06:18
K4NTA-2*   06/09/85  20:57:35
N2WX
cmd:
```

Note that no daytime string is displayed next to N2WX. This indicates that when N2WX was last heard the clock had not been set.

MONITOR ON|OFF **Default: ON**

Parameters:

ON Monitoring of packet activity is enabled.

OFF Monitoring of packet activity is disabled.

If MONITOR is ON and the TNC is not in Transparent Mode, packets not addressed to your TNC may be displayed. The addresses in the packet are displayed along with the data portion of the packet, e.g.:

N2WX>W5FD-3: I'm ready to transfer the file now.

The calls are separated by a ">" and the sub-station ID field (SSID) is displayed if it is other than 0. The MALL, BUDLIST, and LCALLS commands determine which packets are to be monitored. The MCON command controls the action of monitor mode when the TNC is connected. All monitor functions are disabled in Transparent Mode.

The format of the monitor display is controlled by HEADERLN. If you want to see the station addresses on a separate line from the text, you can set HEADERLN ON. MRPT enables monitoring of the digipeater route as well as source and destination addresses for each packet. MSTAMP includes a time stamp with the addresses if DAYTIME has been set.

MRPT ON|OFF Default: ON

Parameters:

ON — Display the stations in the digipeat path for monitored packets.

OFF — Display only the source and destination stations for monitored packets.

This command affects the way monitored packets are displayed. If MRPT is OFF, only the originating station and the destination are displayed for monitored packets. If MRPT is ON, the entire digipeat list is displayed for monitored packets, and stations that have already relayed the packet are indicated with an asterisk.

For example,

WB9FLW>AD7I,K9NG*,N2WX-7:Hi Paul.

This packet, sent from WB9FLW to AD7I, has been relayed by K9NG but not by N2WX-7. With MRPT OFF, the same packet would be displayed as

WB9FLW>AD7I:Hi Paul.

Setting MRPT ON increases the length of the address display, and you may wish to set HEADERLN ON as well to display this information on a separate line.

MSTAMP ON|OFF Default: OFF

Parameters:

ON — Monitored frames are time stamped.

OFF — Monitored frames are not time stamped.

This command enables time stamping of monitored packets. The date and time information is then available for use for automatic logging of packet activity or other computer applications. The date and time are set initially by the DAYTIME command, and the date format is determined by the DAYUSA command.

Setting MSTAMP ON increases the length of the address display, and you may wish to set HEADERLN ON as well to display this information on a separate line.

MYCALL call[-n] Default: NOCALL-0

Parameters:

call	Callsign of your TNC.
n	0 - 15, an optionally specified sub-station ID (SSID).

This command tells the TNC what its callsign is. This callsign will be placed in the FROM address field for all packets originated by your TNC. It will accept frames with this callsign in the TO field and relay frames with this callsign in the digipeat field. MYCALL will also be used for identification packets (see HID and ID).

The default callsign must be changed for proper operation of the protocols. There should never be more than one station with the same callsign (including SSID) on the air at once. The SSID can be used to distinguish two stations with the same Amateur call. The SSID will be 0 unless explicitly set to another value.

NEWMODE ON|OFF **Default: OFF**

Parameters:

ON Switching to data transfer mode occurs at the time of the CONNECT command and return to command mode is automatic at the time of disconnection.

OFF Switching to data transfer mode occurs at time of connection and no return to command mode occurs at disconnection.

The NEWMODE command may be used to select the way the TNC behaves when connections are made and broken.

If NEWMODE is OFF, the TNC will remain in Command Mode after you issue a CONNECT command until a connection is actually established. When the connection is established, the TNC will enter Converse Mode or Transparent Mode, depending on the setting of CONMODE. When the connection is terminated, the TNC remains in Converse or Transparent Mode unless you have forced it to return to Command Mode. This is the same as the behavior of TNC 1 (running version 3.x software) under these conditions.

If NEWMODE is ON, the TNC will enter Converse Mode or Transparent Mode as soon as you issue a CONNECT command, without waiting for the connection to be established. Anything you type will be packetized to be transmitted once the connection is complete. When the connection is broken, or if the connect attempt fails, the TNC will return to Command Mode.

If you have a Bulletin Board program designed to work with TNC 1 you should set NEWMODE OFF if the program relies on the sequence of actions used by TNC 1. Otherwise, you should choose the setting for NEWMODE that seems most convenient to you.

NUCR ON|OFF Default: OFF

Parameters:

ON <NULL> characters are sent to the terminal following <CR> characters.

OFF <NULL> characters are not sent to the terminal following <CR> characters.

This command causes transmission of <NULL> characters (ASCII code $00), producing an effective delay following any <CR> sent to the terminal. The number of <NULL> characters is determined by the command NULLS. This delay is required by some hardcopy terminals. You need to set NUCR ON if your terminal misses one or more characters after responding to a <CR>. If this is the case, you will sometimes see overtyped lines.

NULF ON|OFF Default: OFF

Parameters:

ON <NULL> characters are sent to the terminal following <LF> characters.

OFF <NULL> characters are not sent to the terminal following <LF> characters.

This command causes transmission of <NULL> characters (ASCII code $00), producing an effective delay following any <LF> sent to the terminal. The number of <NULL> characters is determined by the command NULLS. This delay is required by some display terminals. You need to set NULF ON if your terminal sometimes misses characters at the beginning of the line.

NULLS n Default: 0

Parameters:

n 0 - 30, specifying the number of <NULL> characters to send after <CR> or <LF>.

This command specifies the number of <NULL> characters (ASCII code $00) to send to the terminal after a <CR> or <LF> is sent. In addition to setting this parameter value, NUCR and/or NULF must be set to indicate whether nulls are to be sent after <CR>, <LF>, or both. Devices requiring nulls after <CR> are typically hard-copy devices requiring time for carriage movement. Devices requiring nulls after <LF> are typically CRTs which scroll slowly. Extra null characters are sent only in Converse and Command Modes.

PACLEN n Default: 128

Parameters:

n 0 - 255, specifying the maximum length of the data portion of a packet. The value 0 is equivalent to 256.

The TNC will automatically transmit a packet when the number of input bytes for a packet reaches n. This value is used in both Converse and Transparent Modes.

If you perform file transfers, you should experiment with both MAXFRAME and PACLEN. If the link is good, there is an optimum relationship between the parameters set by these commands so that the maximum number of characters outstanding does not exceed the packet receive buffer space of the TNC receiving the data.

NOTE: Although there is no requirement for two TNCs exchanging data to have the same PACLEN value, allowing more than 128 characters of data in a packet may be incompatible with some varieties of TNCs.

PACTIME EVERY|AFTER n Default: AFTER 10

Parameters:

n 0 - 250, specifying 100 ms intervals.

EVERY Packet timeout occurs every n*100 milliseconds.

AFTER Packet timeout occurs when n*100 milliseconds elapse with no input from the computer or terminal.

This parameter is always used in Transparent Mode, and will also be used in Converse Mode if CPACTIME is ON. When EVERY is specified, input bytes are packaged and queued for transmission every n*100 ms. When AFTER is specified, bytes are packaged when input from the terminal stops for n*100 ms. In no case will a zero length packet be produced, and the timer is not started until the first byte is entered. A value of 0 for n is allowed, and causes packets to be generated with no wait time.

PARITY n Default: 3 (even)

Parameters:

n 0 - 3, selecting a parity option from the table below.

This command sets the parity mode for terminal or computer data transfer according to the following table:

n	Parity
0	no parity
1	odd parity
2	no parity
3	even parity

The parity bit, if present, is automatically stripped on input and not checked in Command Mode and Converse Mode. In Transparent Mode, all eight bits, including parity if any, are transmitted in packets. If "no parity" is set and AWLEN is 7, the eighth bit will be set to 0 in Transparent Mode.

PASS n Default: $16 <CTRL-V>

Parameter:

n 0 - $7F, specifying an ASCII character code.

This command selects the ASCII character used for the "pass" input editing command. The parameter n is the ASCII code for the character you want to type in order to include the following character in a packet or text string. You can enter the code in either hex or decimal.

You can use this character to send any character in packets, even though that character may have some special function. For example, suppose you have set COMMAND 3, specifying that <CTRL-C> is your Command Mode entry character. If you use a Bulletin Board program that requires a <CTRL-C> to escape from some operation, you will type

<CTRL-V> <CTRL-C>

to insert a <CTRL-C> character in your packet. Of course, if you do this frequently you would be better off to change your Command Mode entry character.

A common use for the pass character is to allow <CR> to be included in the BTEXT and CTEXT messages. Similarly, you can include <CR> in text when you are in Converse Mode, to send multi-line packets. (The default send-packet character is <CR>.)

PASSALL ON|OFF Default: OFF

Parameters:

ON TNC will accept packets with invalid CRCs.

OFF TNC will only accept packets with valid CRCs.

This command causes the TNC to display packets received with invalid CRC fields. Packets are accepted for display despite CRC errors if they consist of an even multiple of 8 bits and up to 339 bytes. The TNC will attempt to decode the address field and display the callsign(s) in the standard monitor format, followed by the text of the packet.

This mode is not normally enabled, since rejection of any packet with an invalid CRC field is what insures that received packet data is error-free. This mode might be enabled for testing a marginal RF link or during operation under other unusual circumstances.

If you set PASSALL ON and monitor a moderately noisy channel you will periodically see "packets" displayed in this mode, since there is no basis for distinguishing actual packets received with errors from random noise.

Logging of stations heard (for display by MHEARD) is disabled whenever PASSALL is ON, since the callsigns detected may be incorrect.

REDISPLA n Default: $12 <CTRL-R>

Parameters:

n 0 - $7F, specifying an ASCII character code.

This command is used to change the redisplay-line input editing character. The parameter n is the ASCII code for the character you want to type in order to redisplay the current input line. You can enter the code in either hex or decimal.

You can type this character to cause the TNC to retype a line you have begun. When you type the redisplay-line character, the following things happen: First, type-in flow control is temporarily released (if it was enabled). This displays any incoming packets that are pending. Then a <BACKSLASH> character is typed, and the line you have begun is retyped on the next line. If you have deleted and re-typed any characters, only the final form of the line will be shown. You are now ready to continue typing where you left off.

You can use the redisplay-line character to see a "clean" copy of your input if you are using a printing terminal and you have deleted characters. If you have set BKONDEL OFF, deletions are designated with <BACKSLASH> characters, rather than by trying to correct the input line display. The redisplayed line will show the corrected text.

You can also use this character if you are typing a message in Converse Mode and a packet comes in. You can see the incoming message before you send your packet, without cancelling your input.

RESET

This is an immediate command. It resets all parameters to their default settings and re-initializes the TNC.

WARNING: All parameter customizing and monitor lists are lost.

If you want to re-initialize the TNC using the parameter values in battery backed-up RAM, you should turn the TNC off then on again rather than using this command.

RESPTIME n **Default: 12**

Parameters:

n 0 - 250, specifying 100 ms intervals.

This command sets a minimum delay that is imposed on acknowledgment packets. This delay may run concurrently with default wait set by DWAIT and any random wait in effect.

This delay can be used to increase throughput during operations such as file transfer when the sending TNC usually sends the maximum number of full-length packets. Occasionally, the sending TNC may not have a packet ready in time to prevent transmission from being stopped temporarily, with the result that the acknowledgment of earlier packets collides with the final packet of the series. If the receiving TNC sets RESPTIME to 10, say, these collisions will be avoided.

RETRY n **Default: 10**

Parameter:

n 0 - 15, specifying the maximum number of packet retries.

The protocol allows for retries, i.e., retransmission of frames that are not acknowledged. Frames are re-transmitted n times before the operation is aborted. The time between retries is specified by the command FRACK. A value of 0 for n specifies an infinite number of retries. If the number of retries is exceeded, the TNC goes to the disconnected state (with an informative message if not in Transparent Mode). See also the FRACK command.

SCREENLN n Default: 80

Parameters:

n 0 - 255, specifying the screen or platen width, in characters, of the terminal.

This value is used to properly format terminal output. A <CR> <LF> sequence is sent to the terminal at the end of a line in Command and Converse Modes when n characters have been printed. A value of zero inhibits this action.

If your computer automatically formats output lines, you should set SCREENLN 0 to avoid a conflict between the two line formats.

SENDPAC n Default: $0D <CR>

Parameters:

n 0 - $7F, specifying an ASCII character code.

This command selects the character that will force a packet to be sent in Converse Mode. The parameter n is the ASCII code for the character you want to type in order to force your input to be packetized and queued for transmission. You can enter the code in either hex or decimal.

For ordinary conversation, you will probably set SENDPAC $0D and CR ON. This causes packets to be set at natural intervals, and causes the <CR> to be included in the packet.

If you have set CPACTIME ON, you will probably set SENDPAC to some value not ordinarily used (say, <CTRL-A>), and set CR OFF. This will allow you to force packets to be sent, but will not result in extra <CR> characters being transmitted in the text.

START n Default: $11 <CTRL-Q>

Parameters:

n 0 - $7F, specifying an ASCII character code.

This command selects the User Restart character, which is used to restart output from the TNC to the terminal after it has been halted by typing the User Stop character. You can enter the code in either hex or decimal.

The User Stop character is set by the STOP command.

If the User Restart and User Stop characters are set to $00, software flow control to the TNC is disabled, and the TNC will only respond to hardware flow control (CTS).

If the same character is used for both the User Restart and User Stop characters the TNC will alternately start and stop transmission upon receipt of the character.

STOP n Default: $13 <CTRL-S>

Parameters:

n 0 - $7F, specifying an ASCII character code.

This command selects the User Stop character, which is used to stop output from the TNC to the terminal. You can enter the code in either hex or decimal. This is the character you will type to halt the TNC's typing so that you can read text before it scrolls off your display.

Output is restarted with the User Restart character, which is set by the START command.

If the User Restart and User Stop characters are set to $00, software flow control to the TNC is disabled, and the TNC will only respond to hardware flow control (CTS).

If the same character is used for both the User Restart and User Stop characters the TNC will alternately start and stop transmission upon receipt of the character.

TRACE ON|OFF Default: OFF

Parameters:

ON Trace mode is enabled.

OFF Trace mode is disabled.

This command is used to enable the protocol debugging function. When TRACE is ON, all received frames will be displayed in their entirety, including all header information. In normal operation you will probably never need this function; however, if you need to report an apparent software bug, we may ask you to provide trace information if possible.

A trace display will appear in four columns on an 80-column display. Following is an example trace display. For comparison, the frame shown in the trace example would be monitored as follows:

KV7B>CQ,KF7B*:this is a test message

The trace display would look like this:

```
byte  ------------hex display------------ -shifted ASCII--  -----ASCII------
000:  86A24040 40406096 AC6E8440 40609680 CQ     0KV7B  0KF  ..@@@@'..n.@@'..
010:  6E844040 E103F074 68697320 69732061 7B  P.x:449.49.0  n.@@...this is a
020:  20746573 74206D65 73736167 650D      .:29:.6299032.     test message.
```

The byte column shows the offset into the packet of the beginning byte of the line. The hex display column shows the next 16 bytes of the packet, exactly as received, in standard hex format. The shifted ASCII column attempts to decode the high order seven bits of each byte as an ASCII character code. The ASCII column attempts to decode the low order seven bits of each byte as an ASCII character code. In a standard AX.25 packet, the callsign address field will be displayed correctly in the shifted ASCII column. A text message will be displayed correctly in the ASCII column. Non-printing characters and control characters are displayed in both ASCII fields as ".". You can examine the hex display field to see the contents of the sub-station ID byte and the control bytes used by the protocol. Protocol details are discussed in Chapter 9.

TRANS

This is an immediate command. It causes the TNC to exit from Command Mode into Transparent Mode. The current link state is not affected.

Transparent Mode is primarily useful for computer communications. In this mode, the "human interface" features such as input editing capability, echoing of input characters, and type-in flow control are disabled. You may find Transparent Mode useful for computer Bulletin Board operations or for transferring non-text files. See the discussion of Transparent Mode in Chapter 5.

TRFLOW ON|OFF Default: OFF

Parameters:

ON	Software flow control can be enabled for the computer or terminal in Transparent Mode.
OFF	Software flow control is disabled for the computer or terminal in Transparent Mode.

If TRFLOW is ON, the settings of START and STOP are used to determine the type of flow control used in Transparent Mode. If TRFLOW is OFF, only hardware flow control is available to the computer and all characters received by the TNC are transmitted as data. If START and STOP are set to $00, disabling the User Stop and User Restart characters, hardware flow control must always be used by the computer.

If TRFLOW is ON, and START and STOP are non-zero, software flow control is enabled for the user's computer or terminal. The TNC will respond to the User's Restart and User's Stop characters (set by START and STOP) while remaining transparent to all other characters from the terminal. Unless TXFLOW is also ON, only hardware flow control is available to the TNC to control output from the terminal.

TXDELAY n Default: 30

Parameters:

n 0 - 120, specifying 10 ms intervals.

This value tells the TNC how long to wait after keying up the transmitter before sending data. Some startup time is required by all transmitters to put a signal on the air; some need more, some need less. In general, crystal controlled rigs with diode antenna switching don't need much time, synthesized rigs need time for PLL lockup, and rigs with mechanical T/R relays will need time for physical relay movement. The correct value for a particular rig should be determined by experimentation. The proper setting of this value may also be affected by the requirements of the station you are communicating with.

Note that the TAPR TNC 1 and other TNCs using the same firmware interpret n in 40 ms intervals. The value of TXDELAY on TNC 2 will thus be 4 times the value used by a TNC 1 user to give the same delay time.

TXFLOW ON|OFF Default: OFF

Parameters:

ON Software flow control can be enabled for the TNC in Transparent Mode.

OFF Software flow control is disabled for the TNC in Transparent Mode.

If TXFLOW is ON, the setting of XFLOW is used to determine the type of flow control used in Transparent Mode. If TXFLOW is OFF, the TNC will use only hardware flow control and all data sent to the terminal remains fully transparent.

If TXFLOW and XFLOW are ON, the TNC will use the TNC Restart and TNC Stop characters (set by XON and XOFF) to control input from the terminal. Unless TRFLOW is also ON, only hardware flow control is available to the computer or terminal to control output from the TNC.

Note that if the TNC Restart and TNC Stop characters are set to $00, hardware flow control will always be selected regardless of the setting of TXFLOW.

UNPROTO call1 [VIA call2[,call3...,call9]] **Default: "CQ"**

Parameters:

call1 Callsign to be placed in the TO address field.

call2 - 9 Optional digipeater call list, up to eight calls.

This command is used to set the digipeat and destination address fields of packets sent in the unconnected (unprotocol) mode. Unconnected packets are sent as unsequenced I frames with the destination and digipeat fields taken from call1 through call9 options. When no destination is specified, unconnected packets are sent to CQ. Unconnected packets sent from other TNCs can be monitored by setting MONITOR ON and setting BUDLIST and LCALLS appropriately. The digipeater list is also used for BEACON packets (which are sent to destination address BEACON).

XFLOW ON|OFF **Default: ON**

Parameters:

ON XON/XOFF flow control is enabled.

OFF XON/XOFF flow control is disabled and hardware flow control is enabled.

If XFLOW is ON, the computer or terminal is assumed to respond to the TNC Restart and TNC Stop characters set by XON and XOFF. If XFLOW is OFF, the TNC will communicate flow control commands via RTS.

XMITOK ON¦OFF Default: ON

Parameters:

ON Transmit functions are enabled.

OFF Transmit functions are disabled.

When XMITOK is OFF, transmitting is inhibited. All other functions of the board remain the same, in other words, the TNC generates and sends packets as requested, but does not key the radio PTT line.

You might use this command to insure that your TNC does not transmit in your absence if you leave it operating to monitor packet activity. This command can also be used for testing using loopback or direct wire connections when PTT operation is not relevant.

XOFF n Default: $13 <CTRL-S>

Parameters:

n 0 to $7F, specifying an ASCII character code.

This command selects the TNC Stop character, which is sent by the TNC to the computer or terminal to stop input from that device. You can enter the code in either hex or decimal.

This character would ordinarily be set to <CTRL-S> for computer data transfers. If you are operating your station in a Converse Mode and there is some chance that you might fill up the TNC's buffers, you might set this character to <CTRL-G> ($07), which rings a bell on many terminals.

XON n Default: $11 <CTRL-Q>

Parameters:

n 0 to $7F, specifying an ASCII character code.

This command selects the TNC Restart character, which is sent by the TNC to the computer or terminal to restart input from that device. You can enter the code in either hex or decimal.

This character would ordinarily be set to <CTRL-Q> for computer data transfers. If you are operating your station in Converse Mode, and there is some chance that you might fill up the TNC's buffers, you might set this character to <CTRL-G> ($07), which rings a bell on many terminals.

Messages

This section describes the messages your TNC may produce and the circumstances under which they can appear.

MFJ Enterprises, Inc.
Model 1270, TNC 2 Packet Radio
AX.25 Level 2 Version 2.0
Release 1.1.1 - 16K RAM
Checksum $7F
cmd:

This is the sign-on message that appears when you turn on your TNC or when you issue the RESET command. The release number will be updated whenever the firmware is changed. The checksum is a hex number which you can compare against the correct checksum given for the firmware version you are using.

bbRAM loaded with defaults

This message appears along with the sign-on message above if the battery backed-up RAM checksum verification fails at power-on time, causing the TNC to load the default parameters from ROM. (This will be the case the first time you turn on your TNC.)

This message also appears if the TNC loads the defaults in response to the RESET command.

cmd:

This is the Command Mode prompt. When this prompt appears, the TNC is waiting for you to issue a command. Anything you type after this prompt will be interpreted as a command to the TNC. If a monitored packet has been displayed, the prompt may not be visible, even though you are in Command Mode. You can type the redisplay-line character (set by REDISPLA) to retype the prompt.

was

Whenever you change the setting of one of the TNC's parameters, the previous value will be displayed. This confirms that the TNC properly interpreted your command, and reminds you of what you have done.

too many packets outstanding

This message would appear in response to a CONVERS or TRANS command, under special circumstances. If you have previously entered packet data filling the out-

going buffer in Converse Mode or Transparent Mode and then returned to Command Mode. You will be allowed to enter one of these modes when some of the packets have been successfully transmitted.

Command Mode Error Messages

If you make a mistake typing a command to the TNC, an error message will be printed. You may see any of the following messages depending on the type of error you have made.

?bad

You typed a command correctly, but the remainder of the command line couldn't be interpreted.

?call

You entered a callsign argument that does not meet the TNC's requirements for callsigns. A callsign may be any string of numbers and letters, including at least one letter. Punctuation and spaces are not allowed. The sub-station ID, if given, must be a (decimal) number from 0 to 15, separated from the call by a hyphen.

?clock not set

This message appears if you give the command DAYTIME to display the date and time without having previously set the clock. DAYTIME sets the clock if it is given with the daytime parameters, and displays the date and time if it is given without parameters.

?EH

The first word you typed is not a command or a command abbreviation.

?not enough

You didn't give enough arguments for a command that expects several parameters.

?not while connected

You attempted to change MYCALL or AX25L2V2 while in a connected or connecting state.

?range

A numeric argument for a command was too large.

?too long

You entered too long a command line, and the line was ignored. This might happen, for example, if you try to enter too long a message with BTEXT or CTEXT. If you get this message, the previous text entry was not changed.

?too many

You gave too many arguments for a command that expects several parameters. For example, MFILTER can have up to 4 arguments.

```
cmd:MFILTER $1B ,$0C ,$1A ,$03 ,$07
?too many
```

?VIA

This message appears if you attempt to enter more than one callsign for the CONNECT or UNPROTO commands without the VIA keyword.

Link Status Messages

These messages inform you of the status of AX.25 connections your TNC may be involved in. You can always interrogate the link status by giving the CONNECT command without parameters. If you attempt a connection when your TNC is not in the disconnected state, the TNC will display the link status but will take no other action. The following messages appear in response to the CONNECT command.

Link state is: CONNECTED to call1
[VIA call2[,call3... ,call9]]

This display shows the station your TNC is connected to and the digipeater route if any. The callsign sequence is the same sequence you would enter to initiate the connection.

Link state is: DISCONNECTED

No connection currently exists. You may issue the CONNECT command to initiate a connection.

Link state is: CONNECT in progress

You have issued a connect request, but the acknowledgment from the other station has not been received. If you issue a DISCONNE command, the connect process will

be aborted.

Link state is: DISCONNECT in progress

You have issued a disconnect request, but the acknowledgment from the other station has not been received. If you issue a second DISCONNE command, the TNC will go immediately to the disconnected state.

Link state is: FRMR in progress

The TNC is connected but a protocol error has occurred. This should never happen when two TAPR TNCs are connected. An improper implementation of the AX.25 protocol could cause this state to be entered. The TNC will attempt to re-synchronize frame numbers with the TNC on the other end, although a disconnect may result. Connects are not legal in this state, and a disconnect will start the disconnect process.

The TNC will inform you whenever the link status changes. The link status may change in response to a command you give the TNC (CONNECT or DISCONNE), a connect or disconnect request packet from another station, a disconnect due to the retry count being exceeded, an automatic time-out disconnect (CHECK), or a protocol error.

*** CONNECTED to: call1 [VIA call2[,call3...,call9]]

This message appears when the TNC goes from the "disconnected" or "connect in progress" state to the connected state. The connection may be a result of a CONNECT command you issued, or of a connect request packet received from another station.

*** connect request:call1 [VIA call2[,call3...,call9]]

This message indicates that the TNC has received a connect request from another station which it has not accepted. This can happen if you have set CONOK OFF or if you are already connected to another station. When the TNC types this message it also sends a DM packet (busy signal) to the station that initiated the connect request. If the TNC rejects a connect request because you have set CONOK OFF, you can issue your own request to the station that called.

*** DISCONNECTED

This message is displayed whenever the TNC goes to the disconnected state from any other link state. This message may be preceded by a message explaining the reason for the disconnect, below.

*** retry count exceeded
*** DISCONNECTED

This message is given if the disconnect was caused by a retry failure rather than by a disconnect request from one of the stations.

*** <callsign> busy
*** DISCONNECTED

This message indicates that your connect request was rejected by a DM packet (busy signal) from the other station. A MFJ TNC will reject a connect request if CONOK **is** OFF, or if it is already connected to another station, or the connect request is from a station with a totally blank callsign.

frmr frame just sent:
FRMR sent: xxxxxx

The TNC is connected, and a protocol error has occurred. The TNC has sent a special FRMR packet to attempt to re-synchronize frame numbers with the TNC on the other end. The string **xxxxxx** is replaced with the hex codes for the three bytes sent in the information part of the FRMR frame. This message will not appear if your TNC is in Transparent Mode.

FRMR rcvd:

This message is followed by a display of the FRMR packet received in the trace display format. This format is explained in the **TRACE** command entry. This message will not appear if your TNC is in Transparent Mode.

CLKADJ n Default: 0

Parameters:

n 0 - 65535, specifying the correction factor to be applied to the real-time clock routine.

A value of "0" is a special case and means no correction factor will be applied. If the value of **CLKADJ** is non-zero, then the correction factor is calculated as:

$$\text{relative clock speed in \%} = 100 - \left(9.16667 * \frac{1}{n} \right)$$

The real-time clock routine is used to keep track of year, month, day, hour, minute and second as specified in the **DAYTIME** command. It should be noted that the real-time clock is not intended to be your ham-shack reference clock, but is useful for approximate time stamping information.

CONPERM ON|OFF Default: OFF

Parameters:

ON The current connection on the current stream will not be allowed to enter the disconnected state.

OFF The current stream may be connected to and disconnected from other stations.

This command, when switched **ON**, forces the TNC to always maintain the current connection, even when frames to the other station exceed **RETRY** attempts to get an acknowledgment. **RESTART** and power off/on cycling will not affect this connected state.

This command only takes effect when a connection is established. It functions on a stream-by-stream basis when multiple connections are allowed.

It is useful for certain networking applications, meteor scatter and other noisy, less-reliable links, while still allowing connections on other streams to operate normally (automatic disconnect based on **RETRY**, etc.).

CSTATUS

CSTATUS is an immediate command which shows the stream identifier and link state of all ten streams (links), the current input and output streams, and whether or not each stream is "permanent" (see **CONPERM**).

An example fo a display resulting from issuing a **CSTATUS** command is:

```
cmd:CS
A stream - IO Link state is: CONNECTED to 305MLB
B stream -    Link state is: CONNECTED to AD7I P
C stream -    Link state is: DISCONNECTED
D stream -    Link state is: CONNECTED to NOADI via K9NG-2

...

I stream -    Link state is: CONNECT in progress
J stream -    Link state is: CONNECTED to KV7B via NK6K-1
```

The example above shows the A stream is assigned both the input and output streams. The B stream is connected to AD7I "permanently." All other streams' states are shown as they might normally appear with multiple connections.

MYALIAScall[-n] Default: <blank>

Parameters:

call	Alternate identity of your TNC.
n	0 - 15, an optionally specified sub-station ID (SSID).

This command specifies an alternate callsign (in addition to the callsign specified in **MYCALL**) for use as a digipeater **only.**

In some areas, wide coverage digipeaters operators have changed the callsign of their machine to a shorter and (usually) easier to remember identifier. International Civil Aviation Organization (ICAO) airport identifiers, sometimes combined with telephone area codes, have been used.

Use of this command permits **HID** to identify normally with the **MYCALL**-specified callsign yet permit an alternate (alias) repeat-only "callsign."

MFJ TNC 2 Software Release Notes . . . . . Release 1.1.1

RECONNECT call1 [VIA call2[,call3... ,call9]]

Parameters:

call1 **Callsign** of TNC to be reconnected to.

call2 Optional **callsign**(s) of TNC(s) to be digipeate through. As many as eight digipeat addresses can be specified.

RECONNECT is an immediate command. It may be used to change the path through which you are currently connected to a station. It may only be used when your TNC is connected on the current stream to the station you wish to **RECONNECT** to.

Integrity of frames in flight between your station and the **RECONNECT**ed station at the time of **RECONNECT** is not assured.

For details regarding the parameter list, see the **CONNECT** command inthe System Manual.

RESTART

RESTART is an immediate command. It re-initializes the TNC using the defaults stored in bbRAM. The effect of this command is the same as turing the TNC **OFF** then **ON** again.

RESTART does not cause a reset of the parameters in bbRAM. See also the **RESET** command.

STREAMCA ON¦OFF Default: OFF

Parameters:

ON Callsign of other station displayed.

OFF Callsign of other station not displayed.

This command is used to enable the display of the connected-to station after the stream identifier. This is particularly useful when operating with multiple connections allowed. It is somewhat analogous to the use of **MRPT** to show digipeat paths when monitoring.

In the example below, the charaters inserted by enabling **STREAMCAll** are shown in **bold face** type.

¦A:**K4NTA:**hi howie
hello ted how goes it?
¦B:**WA7GXD:***** CONNECTED to WA7GXD
¦Bmust be a dx record. ge lyle
¦Aunreal ted! fl-az no digis!
¦B:**WA7GXD:**big band opening...ge

etc.

The same sequence with **STREAMCA**ll **OFF** would look like the following:

¦Ahi howie
hello ted how goes it?
¦B*** CONNECTED to WA7GXD
¦Bmust be a dx record. ge lyle
¦Aunreal ted! fl-az no digis!
¦Bbig band opening...ge

etc.

Thus, what would have looked like "¦B" now appears as "¦B:<callsign>:". This option is very useful for human operators trying to operate multiple simultaneous connections. It is probably less useful for "host" operations.

Note that, in the first example, the **STREAMSW**itch characters "¦A" and "¦B" with no ":" after them were entered by the operator of the TNC to switch streams for his multiple-connect QSO(s). If you intend to operate multiple connections (as opposed to having your "host" computer operate multiple connections), use of this option is recommended.

STREAMDBL ON¦OFF Default: OFF

Parameters:

ON Double all received STREAMSWitch characters.

OFF Do not "double" received STREAMSWitch characters.

This command is used to display received STREAMSWitch characters by "doubling" them. The example below illustrates this action.

With STREAMDBL on, and STREAMSWitch set to "¦", the following might be displayed from your TNC:

¦¦ this is a test.

In this case the sending station actually transmitted

¦ this is a test.

The same frame received with STREAMDBL OFF would be displayed as:

¦ this is a test.

When operating with multiple connections, this is useful for differentiating between STREAMSWitch characters received from other stations and STREAMSWitch characters internally generated by your TNC.

NOTE: The STREAMSWitch character must NOT be one of the stream letters (A - J) for this command to function properly.

STREAMSW n Default: $7C < ¦

Parameters:

n 0 - $FF, usually specifying an ASCII character code.

This command selects the character used by both the TNC and the user that a new "stream" (connection channel) is being addressed.

The character can be PASSed in CONVERS mode. It is always ignored as a user-initiated stream switch in TRANSPARENT mode, and flows through as data. This means that the outgoing stream cannot be changed while "on line" in TRANSPARENT mode (you must escape to COMMAND mode to switch streams).

For further usage of this character, see teh STREAMDBL and STREAMCA commands.

TRIES n Default: NONE

Parameters:

n 0 -15, specifying the current **RETRY** level on the currently selected input stream.

This command is used to retrieve (or force) the count of "tries" on the currently selected input stream.

When used with no argument: if the TNC has an outstanding unacknowledged frame, it will return the current number of tries; if the TNC has no outstanding unacknowledged frames, it will return the number of tries required to obtain an acknowledgment for the previous frame.

If **RETRY** is set to 0, the value returned by issuing a **TRIES** command will always be 0.

This command is useful for obtaining statistics on the performance of a given path or channel. It should be especially useful for automatic optimizing os such parameters as **PACLEN** and **MAXFRAME** by computer-operated stations, such as automatic message forwarding stations using less-than-optimum paths (noisy HF or satellite channels, for example).

When used with an argument, **TRIES** will force the "tries" counter to the entered value. Use of this command to force a new count of tries is **NOT** recommended.

USERS n Default: 1

Parameters:

n 0 - 10, specifying the number of active connections that may be established to this TNC by other TNCs.

USERS affects only the manner in which incoming connect requests are handled, and has no effect on the number or handling of connections you may initiate with this TNC.

For example,

USERS 0 allows incoming connections on any free stream
USERS 1 allows incoming connections on stream A only
USERS 2 allows incoming connections on streams A & B
USERS 3 allows incoming connections on streams A, B & C

and so on through **USERS 10**.

HARDWARE

This chapter includes detailed specifications and a functional description of the hardware design of the TAPR TNC 2.

TNC 2 Specifications

Processor CMOS Z-80A

Clocks Processor master clock input frequency: 2.4576 MHz jumper selectable to 4.9152 MHz (requires use of Z80B CPU, Z80B SIO/O and 200 nSec EPROMs).

User Port Clock: Switch selectable at 16x baud rate.

MF-10 switched-capacitor filter clock: 153.6 kHz.

Memory All memory in industry-standard JEDEC Byte-Wide sockets.

Standard complement of ROM: 32k = 1 x 27C256

Standard complement of RAM: 16k = 2 x 6264LP

ROM complement jumper selectable for 8k-, 16k-, or 32k-byte parts. Memory can be expanded to 32k ROM and 32k RAM using currently available parts.

Serial Port Z8440 SIO/O port B configured as UART plus low-power TTL-to-RS-232C signal level interface.

Baud rates supported: 300, 1200, 2400, 4800 and 9600.

Standard female DB-25S (DCE) RS-232C connector.

Modem Input demodulation filter: MF-10 switched-capacitor filter permits use with unmodified FM transceivers. Easily configured for SSB via replaceable DIP headers.

Demodulator: XR2211 PLL demodulator circuit plus related components to receive up to 1200 baud 1200 Hz/2200 Hz FSK modulation.

Modulator: XR2206 modulator circuit plus related components to produce 1200 baud 1200 Hz/2200 Hz phase coherent FSK.

All important passive modem components mounted on plug-in DIP headers for ease in substitution and modification.

Built-in modem calibration system requires no additional equipment.

An external modem may be attached via a single connector which completely bypasses the on-board modem.

Support for an external modem tuning indicator is provided via a separate connector.

bbRAM Non-volatile storage of all important operating parameters is accomplished by using a battery backed-up system for the entire 16k bytes of system RAM.

Protocol AX.25 Level 2 is supported. Pre-Version 2.0 support is compatible with earlier TAPR TNCs running 3.x software. Full support of Version 2.0 protocol is provided. Full duplex radio link operation is supported.

Operating Modes Command Mode: accepts commands via user port.

Converse Mode: accepts digital data, transmits and receives packets, permits terminal editing features (character delete, line delete, input packet delete, output packet delete and redisplay input) via special characters trapped by the TNC. Escape to command mode via special character or BREAK signal. Optional use of packet completion timer as in Transparent mode.

Transparent Mode: accepts digital data, transmits packets via packet completion timer or buffer full only, and receives packets. No local editing features permitted. Escape to command mode via specially timed character sequence or BREAK signal.

Power Required +10 to +15 volts DC at less than 200 mA.

General Description

The TNC 2 is based on the Zilog Z80 (tm) family of microprocessor components. All parts used in the TNC should be readily available to the Radio Amateur.

Major electronic devices in the TNC include a CPU (Central Processing Unit) for controlling the TNC and a SIO (Serial Input/Output) chip for providing serial interface ports to the on-board modem (which connects to your radio) and to the RS-232C serial terminal port (which connects to your computer or terminal).

The TNC also includes two types of memory. ROM (Read Only Memory) stores the program that tells the TNC how to implement the AX.25 protocol. Battery backed-up RAM (Random Access Memory) provides a scratch-pad area for temporary data as well as non-volatile storage for operating parameters such as your station call sign. The battery back-up feature enables the TNC to "remember" these values when power is off so you don't have to enter them every time you want to operate.

Other integrated circuits are used for functions including clock oscillator, baud-rate generator, memory-space decoder, power supply and voltage inverter, clock recovery, transmit watch-dog timer and modem. Refer to the schematic diagram while reading the following circuit descriptions.

Detailed Circuit Description

Oscillator

U10a, U10b, U10c, R46, R47, R48, C24, C47, C51, and Y1 provide an accurate crystal-controlled oscillator for system timing.

R48 forces inverter U10a into its linear region and provides a load for crystal Y1. C47 provides an adjustable reactive element to allow the oscillator's frequency to be precisely set (this precision is not normally required). Inverter U10c buffers the clock for additional stability before driving additional dividers.

R46 is used to bias "HCT" logic to the proper levels for best oscillator operation; it is not necessary if U10 is an "HC" logic element.

Dividers and Baud-rate Generator

U10e, U10f, U4a, U4b, U1, and SW2 provide clock outputs derived from the oscillator. Only one of the first 5 switches (terminal baud rate) and only one of the last 3 switches (RF baud rate) should be on at any time.

Inverter U10f provides buffering and isolation between the divide-by-two output of counter U4a and the capacitive load presented by the CPU (U22) and the SIO (U21). U10f's input may be at 2.4576 MHz (standard) or 4.9152 Mhz via jumper JMP2. Operation at the faster clock may require the use of higher-speed parts. Capacitors C59 and C60 are used to slow the edges of the outputs of U4a, and capacitor C61 is used to slow the edges from U10f, helping to reduce RFI.

Counter U1 is a multiple-stage divide-by-two circuit that divides the signal at its input many times. This allows you to select, via switch SW2, the desired signaling (baud) rate to be used for your computer or terminal as well as the radio channel baud rate.

The output from counter U1 at pin 12 provides a real-time clock interval signal for the SIO. During normal operation, the SIO will be programmed to interrupt the CPU on every transition of this 600 Hz signal. This interrupt occurs 1200 times a second, and is used for protocol and calibration timing functions.

Inverter U10e buffers the radio port "16x" baud-rate signal in case it is routed, via modem disconnect J4 pins 11 and 12, to an external modem. If this buffer were not included, reflections from the distant termination might cause counter U1 to generate count errors.

Counter U4b provides a properly scaled clock for the transmit NRZ to NRZI encoder (see Serial Interface, below).

CPU Complex

EPROM U23 provides system ROM for program storage. Selector U12a acts as a ROM decoder, mapping the ROM into the CPU's memory address space beginning at address 0.

Static RAMs U24 and U25 provide system RAM for temporary scratch-pad storage, message buffers, etc. Also, because the RAM is backed up by a battery and will not lose its contents when the main power is removed, it is used to provide semi-permanent storage of user-supplied information (such as your call sign). Selector U12b acts as a RAM address decoder, with RAM starting at address 8000 hex.

JMP12 allows selection of 16k bytes (using 2 x 6264 8k RAMs) or 32k bytes (using 1 x 43256 32k RAM)) of RAM.

The sections of CMOS switch U13 are used to insure that the RAM is not selected when main power is removed. This ensures that the contents of the RAM are not accidentally scrambled as the CPU loses power; it also ensures that the RAM is in the "power-down" state for minimum battery power consumption.

Serial Interface

Serial Input/Output (SIO) device U21 provides two channels of serial I/O.

The B SIO channel is used for the computer or terminal interface. Operational amplifier sections U3a, U3b, and U3d act as RS-232C drivers while Schmitt trigger inverters U9a and U9b act as RS-232C receivers. These circuits consume less power than conventional RS-232C drivers and receivers.

The A SIO channel is used for the radio/modem interface and is normally operated as a full duplex HDLC channel for compatibility with the AX.25 protocol specification. Latch U5 and ROM U6 provide a "state machine" for recovering the clock from the received NRZI data. The state machine also converts NRZI data to NRZ for the SIO. Inverter U9c and flip-flop U11a provide NRZ to NRZI conversion for the transmit side of the radio channel. This conversion between NRZ formatted data and NRZI formatted data is necessary because the AX.25 protocol specification requires NRZI operation while the SIO is only capable of NRZ. Jumper JMP11 may be used to bypass the NRZ --> NRZI conversion for use with external modems, if required.

Watch-dog Timer

Inverters U7c, U7d, U7e, and Q10 provide a "watch-dog" timer on the transmit key line to ensure that the transmitter does not remain keyed for more than about 30 seconds if the TNC fails. This allows you to leave a station (such as a remote digipeater) on-the-air and unattended without much chance of having a malfunction "lock up" the packet channel. This also helps ensure compliance with FCC regulations regarding unattended station operation.

Jumper JMP4 is provided for testing purposes. When JMP4 is installed, timing capacitor C31 is shunted, disabling the watch-dog timer.

Modem

U16, an XR2206, is a Frequency Shift Keying (FSK) modulator that generates an audio data signal for use by the radio transmitter. Two tones are used, one for each digital level, and these tones may be calibrated via trimpots R77 and R78 in conjunction with the on-board calibration support circuitry and software. When the transmitter is not being keyed, transistor Q9 is switched on, thus preventing U16 from producing tones. This allows you to leave a microphone connected to your packet transmitter for voice operation. DIP header U15 contains the three passive components which determine the operation of the modulator, as well as R83, the PTT watch-dog circuit timing resistor.

R76 is used to set the tone output level to the transmitter.

U20, an XR2211, provides a Phase Locked Loop (PLL) FSK demodulator. It converts the received audio FSK signals into digital data at standard logic levels. This data is sent to the state machine clock recovery and NRZI to NRZ format conversion circuits. R79 is used to calibrate the PLL demodulator's free-running frequency which is set midway between the FSK tones being received. This tone is measured by the calibration software and the output signal produced by U8a and U8b, which is a frequency doubling shaping circuit. DIP header U19 contains all passive components which determine the behavior of the demodulator, with the exception of R73 and C45, the DCD detection filter. These two components normally do not need to be changed when the modem is reconfigured. However, in the event the TNC will be dedicated to 300 baud operation, the DCD detection filter at U18 pin 3 should have its time constant increased by a factor of 4, by increasing C45.

U18, an MF-10 switched capacitor filter, provides audio amplitude and phase equalization to shape the received audio signal before it goes to the XR2211 demodulator. This equalization helps the demodulator properly process signals by correcting some of the distortion that many FM radios introduce. By replacing the components on DIP header U17, this filter may be configured at will.

Power Supply

Regulator Q3 and associated components provide a +5 volt regulated output for the TNC digital logic circuitry. In order to reduce conducted RFI from the digital power source, series inductor L1 is provided. Transistor Q4, in conjunction with CMOS inverters in U14, provides a "power failure" circuit for the battery-backed RAM chips to ensure that RAM is in the "power-down" state when the main power is

removed. In addition, this circuit provides the main power-on reset signal via U7f.

Transistors Q5 and Q6 are used to isolate the battery from the +5 volt line when main power is available. R33 protects the lithium battery from overload conditions and provides a convenient means of monitoring battery current drain when the TNC is switched off. JMP5 provides a means of disconnecting the lithium battery for TNC 2 maintenance.

U2 and associated components form a charge-pump voltage inverter which generates an unregulated negative supply for the RS-232C drivers.

U3c, Q2, CR6 and surrounding components provide a regulated source of -5 volts for the modem chips.

The modem power sources and ground are isolated from digital logic switching noise by inductors L2 and L3.

RS-232C Handshaking Protocol

The CTS, DSR and DTR lines of the RS-232C port (J1) are used for hardware "handshaking" protocol to control the flow of data between the terminal (DTE) and the TNC (DCE).

The TNC always asserts (makes true) Data Set Ready (DSR) on J1 pin 6 via resistor R16. Thus, whenever the TNC is powered up, it signals to the terminal connected to J1 that the TNC is "on line."

The terminal indicates it is ready to receive data from the TNC by asserting its Data Terminal Ready (DTR) output, J1 pin 20. The TNC will send data when it has data to send and DTR is asserted. If the terminal is not ready to receive data, it should negate (make false) DTR to the TNC. Thus, data flow from the TNC to the terminal is controlled by the use of the DTR line. The state of the DTR line is ignored by the software if "software flow control" is enabled in this direction.

The TNC asserts its Clear To Send (CTS) output, J1 pin 5, whenever it is ready to receive data from the terminal. If the TNC's buffers fill, it will negate CTS, signaling the terminal to stop sending data. The TNC will assert CTS when it is again ready to receive data from the terminal. Thus, data flow from the terminal to the TNC is controlled by the use of the CTS line. The CTS line is always asserted if "software flow control" is enabled in this direction.

Some serial I/O ports do not implement CTS, DTR and DSR handshaking. If these pins are not connected at the termi-

nal end, they will be pulled up (and thus asserted) by resistors at the TNC end. However, a non-standard serial connector may use some pins for other purposes, such as supplying power to a peripheral device, so be sure that your system either implements the CTS, DTR and DSR handshake or has no connections to these pins of J1 whatsoever. Note that reference to RS-232C "compatibility" or the presence of a DB-25 type connector does not guarantee that you have a full RS-232C serial port!

The TAPR TNC 2 supports most standard baud rates from 300 through 9600. The port supports standard parity options as well as 7- or 8-bit character lengths. Setting these terminal parameters is discussed in Chapter 4.

If you want to interface your TNC with a device configured as DCE, such as a telephone modem or another TNC, a so-called "null modem" cable may be constructed to interchange the data and handshake signals. See for example Byte, February, 1981, page 198.

Jumper Functions

The following table lists the function of each jumper on the TNC 2 PC board. The default positions for JMP7, JMP8, JMP9, and JMP10 are necessary for normal operation to occur.

Jumper	Position	Function
JMP1	ON	!DCD (RS-232C) stays on
	OFF (default)	!DCD reflects connect status
JMP2	LEFT	4.92 MHz CPU clock
	RIGHT (default)	2.46 MHz CPU clock
JMP4	ON	disable Tx watch-dog
	OFF (default)	enable Tx watch-dog
JMP5	ON	Lithium battery connected
	OFF (default)	Lithium battery disconnected
JMP6	ON (default)	U23 is type 27256
	OFF	U23 is not type 27256
JMP7	ON	analog loopback mode
	OFF (default)	normal modem operation
JMP8	ON	demodulator enabled
	OFF (default)	demodulator calibrate

(continued)

Jumper	Position	Function
JMP9	TOP	calibrate U16 tones
	MID	calibrate U20 tone
	LOW	bypass state machine
	OFF (default)	normal modem operation
JMP10	ON	digital loopback mode
	OFF (default)	normal modem operation
JMP11	LEFT	transmit data NRZ
	RIGHT (default)	transmit data NRZI
JMP12	TOP (default)	16k RAM (2 x 6264)
	BOTTOM	32k RAM (1 x 43256)

Modem Disconnect - J4

The modem disconnect, J4, on the TNC 2 PC board is provided for using an external modem with the TNC. This allows use of higher-speed modems, such as 9600 baud, or more sophisticated, higher-performance modems for OSCAR or other uses.

The following information is primarily for those who wish to interface external modems to the TNC. Familiarity with modem and serial data channel terms is assumed.

A physical connector for J4 is not supplied with the TNC. Any standard 20-pin header for use with IDC cable connectors should work. Suitable parts are the Ansley 609-2027, the 3M 3428-6202, etc.

When installing the connector, be sure to line up the marked pin (pin 1) of the header with the lower edge of the PC board nearest U21. Cut the traces on the bottom of the TNC PC board, with a sharp knife or Moto-Tool, between each pair of adjacent pins of J4. To use the on-board modem, install push-on jumper links across these same pin pairs.

The signals used at connector J4 are at standard TTL interface levels. A TTL high, or 1, is greater than +2.4 volts but less than +5.25 volts. A TTL low, or 0, is less than 0.8 volts but greater than -0.4 volts. **DO NOT connect an RS-232C level modem directly to J4!**

> **NOTE:** The modem disconnect is similar, but not identical to that used in TNC 1. Be very careful about interfacing an external modem using the same cabling you may have prepared for use with TNC 1!

The connector pin-outs are as follows.

Pin 1 Carrier Detect Input

This pin tells the SIO radio port that a valid data carrier has been detected. It should be pulled high when no carrier is detected and low when a carrier is present. This line must be implemented unless the software release notes indicate otherwise. It is normally jumpered to pin 2 when the on-board modem is used.

Pin 2 Carrier Detect Output

This pin is an output from the the on-board modem and satisfies the requirements outlined for pin 1 above. It is normally jumpered to pin 1 when the on-board modem is used.

Pin 3 SIO Special Interrupt Input

This signal is routed to the radio port DCD input pin on SIO U21. This signal is normally used during modem calibration. It may also be used for other purposes; if so, these functions will be listed in the software release notes. This pin is normally jumpered to pin 4 when the on-board modem is used.

Pin 4 SIO Special Interrupt Generator Output

This signal is an output from the on-board modem. It is normally used for modem calibration only. If it is used for other functions, they will be stated in the software release notes. This pin is normally jumpered to pin 3 when the on-board modem is used.

Pin 5 SIO RTS Output

This signal is used for transmitter activation. The SIO will pull this output low when the TNC wants to transmit; otherwise it will remain high. This pin is normally jumpered to pin 6 when the on-board modem is used.

Pin 6 Transmitter Key Input

This signal is an input to the on-board modem. It activates the PTT pin of the radio connector via the watch-dog timer. It should be left high and pulled low only when transmission is desired. This pin is normally jumpered to pin 5 when the on-board modem is used.

Pin 7 CONNECT Status Output

This pin is an output from the SIO. It is normally low and goes high only when the TNC is in the connected (error-free) mode with another packet station. Its status is monitored via the CON LED.

Pin 8 Unacknowledged Packets Pending Status Output

This pin is an output from the SIO. It is normally low and goes high only when this TNC has unacknowledged packets in its transmit buffer. Its status is monitored via the STA LED.

Pin 9 CTS Input

This pin is an input to the SIO. It is high when the attached modem is not ready to accept data, and low when the attached modem is ready to accept data. The TNC will not attempt to send data when this pin is high. This pin is normally jumpered to pin 10 when the on-board modem is used.

Pin 10 Transmitter Key Input

This pin is physically tied to pin 6, above. It is used in conjunction with pin 9, above, to allow the TNC to use the on-board modem whenever the transmitter is activated.

Pin 11 Transmitter Clock (16x) Input

This pin is tied to the NRZ-to-NRZI converter, which expects a clock signal at 16 times the desired radio port data rate, e.g., 4800 Hz for 300 baud. This pin is normally jumpered to pin 12 when the on-board modem is used.

Pin 12 Transmitter Clock (16x) Output

This pin is tied to the radio baud rate switch network. It provides a clock at 16 times the desired radio port data rate. This pin is normally jumpered to pin 11 when the on-board modem is used.

Pin 13 Receive Clock Input

This pin is tied to the SIO receive clock input pin. It expects a clock at the desired data rate (1200 Hz for 1200 baud), of the proper phase relationship to the received data. This pin is normally jumpered to pin 14 when the on-board modem is used.

Pin 14 Receive Clock Output

This pin is the received data clock signal derived from the NRZI-to-NRZ state machine. This pin is normally jumpered to pin 13 when the on-board modem is used.

Pin 15 TNC 2 Ground Reference

This pin ties to the TNC digital ground system, at the SIO.

Pin 16 No Connection

This pin is unused.

Pin 17 Receive Data Input

This pin is the received data input to the NRZI-to-NRZ state machine. This pin is normally jumpered to pin 18 when the on-board modem is used.

Pin 18 Receive Data Output

This pin provides receive data from the on-board modem. This pin is normally jumpered to pin 17 when the on-board modem is used.

Pin 19 Transmit Data Output

This line is the NRZ or NRZI (depending on the state of JMP11) data output. This pin is normally jumpered to pin 20 when the on-board modem is used.

Pin 20 Transmit Data Input

This input line accepts data to be be transmitted by the modem. This pin is normally jumpered to pin 19 when the on-board modem is used.

If you elect to use an off-board modem, be sure to properly shield the interconnecting cables for RFI protection.

Tuning Indicator Interface - J3

In order to facilitate communications on HF and OSCAR, the TAPR TNC 2 includes a connector for attaching a tuning indicator. The attached unit may range from an oscilloscope to a specialized, LED-style unit such as the one described in the June 1984 PSR. Refer to the Exar Application Note referenced in the Bibliography for details on functions of the XR2211 signals available on this connector. A mating

connector assembly (shell and 5 pins) may be obtained from TAPR.

NOTE: Exercise caution when using this connector. **Be especially aware that the TNC 2 modem circuitry is referenced to +5 volts and -5 volts, not +12 volts and 0 volts!**

The connector pinouts are as follows.

Pin 1 -5 volts (Common)

This pin is the on-board modem's analog common. It is not ground, but -5 volts. This pin should not be used to sink appreciable currents or excessive noise may be introduced into the modem, reducing its performance.

Pin 2 Loop Data Filter Output

This pin is connected to the output of the XR2211 PLL data filter (U20 pin 8). It is a high-impedance source, and care should be exercised to ensure that no extraneous signals or low-impedance loads are attached.

Pin 3 Demodulator Reference Voltage

The internal XR2211 data comparator reference voltage is available on this pin. By comparing this voltage with the signal on J3 pin 2, correct tuning may be accomplished. As above, this pin must be carefully shielded from noise, and has a relatively high internal impedance.

Pin 4 Data Carrier Detect

This pin is an open-collector output that goes near -5 volts (J3 pin 1) when valid data is not present. It is pulled to +5 volts by R74 when valid data is detected.

Pin 5 +5 Volts

This pin is a source of +5 volts DC. It should not be used to source more than a few milliamperes of current or degradation of the on-board modem's weak-signal performance may result.

TROUBLESHOOTING

WARNING: Never remove or insert an IC with power on!

Your TAPR TNC 2 is a complex piece of electronic equipment. Servicing must be approached in a logical manner. The best preparation for troubleshooting is to study the detailed hardware description in Chapter 7. While it is not possible to present all possible problems, symptoms and probable cures, this section of the manual will give direction to troubleshooting based on our experience.

General Tests

In most cases we have found that careful visual inspection combined with simple measurements generally reveals the problem. The most useful single instrument for troubleshooting is a good DVM that can read AC and DC volts, and can non-destructively test resistance while the ICs are still in their sockets.

While a number of checks may be made without the aid of an oscilloscope, you will need one to check signals at various points on the board if you fail to locate the problem by visual means or with a meter. Be very careful about shorting pins on ICs when applying meter or scope probes to the board. It is a good idea to attach a secure ground lead to the meter or scope, one that won't accidentally short across components on the board. A good place to pick up this ground is on the threads of the screw that mounts regulator Q3 to the printed-circuit board.

Step 1: Power Supply

The first thing to check in any malfunction is the power supply. Check the power supply levels at the outputs of the voltage regulators (Q2 and Q3) as well as the output of the inverter (U2). Are they close to their nominal values? Do all the ICs in the suspected area have the proper voltage on their power pins? Is there excessive ripple in any of the DC voltage lines? If so, check the regulator and associated components, working backwards toward the input power switch. If the voltage is low, in conjunction with a hot regulator, suspect a short circuit on the board.

If the problem is in the -5 volt supply, work backwards from Q2's collector (also at U16 pin 1), which should be -5 volts regulated, to the junction of C9 and CR2 (-**V** unregulated). If no voltage appears at -**V**, then U2 or a related component may be at fault. Verify that U2 is oscillating by looking at the wave form at U2 pin 5. If -**V** is more negative than -7 volts (i.e., normal), but the -5 volt regulated voltage is wrong, check the negative regulator components U3, Q2, R5, R6, R7, R8, CR6, C10, C11 and C158. If both -**V** and the -5 volt regulated voltage are wrong, look for shorts.

Step 2: Obvious Problems

Look for any unusual physical symptoms. Have you installed any ICs the wrong way? This is almost guaranteed to ruin the IC and produce a high current through it, detectable by the IC's high temperature. Are any components discolored? Does something smell burnt? Do any of the parts seems excessively warm? If you have never had your fingers on operating digital integrated circuits before you may erroneously conclude they are too hot when they are actually operating normally. In general their normal temperature will be well below the boiling point of water, but you may not want to keep your finger on them very long.

Step 3: Assembly Problems

Carefully inspect the PC board and component installation. Are any cold solder joints present? Is a metal screw shorting to the board anywhere? Are all ICs firmly seated in their sockets? Are any IC leads tucked under the chip or otherwise bent in such a manner that they aren't making proper contact with the IC socket? (This is a **very** common error, accounting for most problems!)

Inspect the diodes and electrolytic capacitors for proper installation. Are the diode cathodes pointing the correct way? Are the negative ends of the electrolytic capacitors pointing the correct way?

Step 4: Cabling Problems

Inspect the interconnection cabling. Does it work on another TNC? Has the radio and/or terminal been successfully used on packet with this or another TNC? Are all the connections tight? Has the cable frayed or broken?

Specific Symptoms

While the steps described above may seem obvious, careful inspection often will point to the problem or give significant clues as to the probable area of the TNC most suspect. After the above inspection has been completed and apparent problems dealt with, it is time to proceed to more specific analysis.

Symptom: TNC appears dead

If the TNC powers up with the PWR, STA, and CON LEDs lit, followed by STA and CON extinguishing a second or so later, the processor is working and the software is probably working correctly. You should suspect the terminal port at this point. Check all connections and verify the logic levels according to the terminal interface troubleshooting section in this chapter.

Oscillator and Reset Circuits

If no LEDs wink during the reset cycle the problem may be more serious. Check to see that the crystal oscillator is working and that an "M1" signal (154 kHz square wave, 0 to +5 volts) is coming from U22 pin 27. The crystal oscillator input to the processor (U22) is pin 6. The input clock should be a (possibly distorted) square wave signal. Verify that the clock input at pin 6 of U22 is running at the correct frequency (near 2.4576 MHz).

Verify that the battery backed-up RAM protection circuit, composed of Q4, U14 and associated devices, is going to +5 volts at U14 pin 6 after input power is applied. This signal enables normal operation of U24 and U25. There should be a logical low on the output of U7 pin 12 coincident with the application of power and lasting for a few hundred milliseconds. Without this RESET signal, the Z80 probably won't start up properly.

Digital Logic Lines

Remember that all the logic circuits operate at standard TTL levels (a "low" is less than +0.8 V and a "high" is greater than +2.4 volts), and all digital inputs and outputs switch between these two levels. Thus, if you see logic signals switching between 0 and, say, 1 volt, you can be sure there is a problem (usually a short). On the other hand, do not mistake switching transients on digital logic lines for improper operation -- these show up as ringing and other distortions.

Verify that there is activity on the control bus READ and

WRITE lines, the 3 CHIP-ENABLE lines on the memories (U23-25, pins 20), the IOREQ line on U21 pin 36, and the INT line on pin 16. Each of these lines should show activity, and if any line is quiet this is a sign of trouble.

Logic lines that show no activity may often be traced to a short on the pc board, probably due to a solder splash or bridge.

Address and data line shorts may also show up as lack of activity on the control bus lines, especially the chip selects. Check each of the 16 address and 8 data lines for activity. Any lines showing a lack of activity are not operating properly.

If you suspect problems with address or data lines, try removing all the memory chips. Each address and data line will now show a distinct pattern. The address lines should be (possibly distorted) square waves whose periods increase by a factor of two on successive lines as you step line by line from A0 to A15.

If you decide to use an ohmmeter to check for shorted lines, use a low voltage/low current test instrument. (Most modern DVMs are fine for this.) If in doubt, remove any ICs connected to the lines you are measuring. If you suspect a short, check the high density areas of the PC board for the problem. In most cases the short will be found there. It is very unlikely that the PC board itself will have a short, as every board shipped by MFJ has been electrically tested for shorts and opens on a commercial "bed-of-nails" board tester prior to acceptance by MFJ.

Symptom: Modem won't calibrate or key transmitter.

Double check the placement of parts on each DIP header assembly, measuring resistor values and checking for shorts between adjacent components and for shorted capacitors.

Calibration of the demodulator's 1700 Hz tone and the modulator's 1200 Hz and 2200 Hz tones is done in software by setting the specified device to generate the frequency in question and routing the signal to U21 pin 22 where it causes interrupts to the Z80 microprocessor. The software calibration routine then examines the ratio between this interrupt rate and a fixed 1200 Hz interrupt generated by a 600 Hz square wave at U21 pin 29, then sets the STA and CON LEDs appropriately. In the case of the 1700 Hz tone, it is first passed through U8, acting as a frequency doubling Schmitt trigger. The input signal to the Schmitt trigger is a modified sawtooth wave. In the case of the modulator, the

signal presented to U21 pin 22 comes directly from U16 pin 11, and should be a reasonable square wave.

Troubleshooting improper calibration amounts to checking for proper signals at U21 and following up any improper signal. If the calibration signal is present, but you cannot successfully calibrate the frequency, you may have an out-of-spec frequency determining component. Check the values of the appropriate passive components. Also, check the placement of jumpers! As a last resort, check the signal frequency with a frequency counter. Note that, due to frequency jitter while calibrating the demodulator, the STA and CON LEDs may blink somewhat even when the 1700 Hz demodulator frequency is correct.

If the transmitter doesn't key, the problem may be in the watchdog timer, U7, or the PTT transistor, Q10. Check especially for an open timing capacitor C31 or a bad solder connection associated with R83 in header U15.

Symptom: Uncopyable transmitted or received packets

If no one seems able to decode your packet transmissions, it is often the case that your transmitter is being overdriven. The solution is to reduce the drive level via trimpot R76. Note that direct connection to typical microphone inputs requires R76 to be turned to near the minimum signal position to produce sufficiently low signal levels.

If you are having problems hearing other stations, the demodulator circuitry associated with U20 may be at fault. Check the center frequency of the VCO in U20 using the 1700 Hz calibration procedure. Working in the direction of flow of the input signal from the radio, verify that it is being passed through to pin 2 of U20, the input pin. The signal there should be above 50 mV and below 3 V peak-to-peak for proper operation of the demodulator. It should be relatively clean, although a few tens of millivolts of noise is normal, and the signal amplitude should not change by more than about 25% between high and low tones.

Note: Make sure that JMP 8 is on.

Terminal Interface Troubleshooting

If you can't get the TNC to sign on and accept data from your terminal or computer, the problem may be in the RS-232C interface. The troubleshooting guide below is provided as an aid to help in resolving problems that may be related to the RS-232C port.

Symptom: TNC won't sign on to the terminal.

If you find the TNC won't send data to your terminal, one of the first things to do is to verify that the DTR line at pin 20 of J1 is not being held low. If the software flow control option is disabled, the TNC will not send data to the terminal unless its DTR is asserted. If the terminal does not implement the DTR/CTS protocol, the DTR/CTS lines (pins 20 and 5 on J1) should remain unconnected.

Verify that the voltages on the TNC are correct. If the TNC is in otherwise good condition, check the following pins on the SIO, U21 (Z8440). Pin 23 should be TTL low (between 0 and +0.8 volts). If this voltage is incorrect, check the voltage at U9 pin 3 and verify that it is greater than +3 volts. If this voltage is correct, U9 or the traces around it may be bad. If this is not the problem, disconnect the terminal and check it again. If this doesn't help, U9, R20 or R22 may be at fault.

If the above checks are ok, observe pin 26 of U21 with an oscilloscope and cycle the power switch on the TNC. Transitions on this pin shortly after reset indicate that the TNC is sending data. Verify that transitions are also present on U3 pin 1. If these tests fail, the fault could be with U13, R25, R26, U21, J1, the attached cable or faulty soldering (shorts, cold joints, etc.)

Symptom: The TNC appears to be signing on but only gibberish is printed on the terminal.

This indicates that some combination of the data rate (baud rate), parity option, or number of start and stop bits are not set the same at the TNC and at the terminal. If possible, set your terminal to 300 baud. Also verify that the terminal is set for seven data bits, space parity, and 1 stop bit. These are the default settings stored in EPROM. For 300 baud, set DIP switch 1 ON (up) and set switches 2 through 5 OFF (down). Be sure only one of these 5 switches is up! Perform a hard reset by the power switch OFF then ON (out then in). The sign on message should appear.

If the TNC still prints gibberish, verify that the terminal is set to 300 baud and do a power off then on cycle on both the TNC and terminal. If the message still fails to appear try troubleshooting with an oscilloscope, looking first at the TXD pin (pin 26) of U21 (Z8440), then at the x16 baud rate clock (4800 Hz at 300 baud) on pin 27 of U21.

Symptom: The TNC signs on OK but won't accept commands.

After the TNC signs on, try giving it a command such as **MYCALL** or any other command. If the default settings are in effect, it will attempt to echo each character you type back to the screen. If it doesn't echo, be sure that U21 pin 23 has a voltage level between 0 and +0.8 volts on it. The voltage on U9 pin 3 should be greater than +3 volts. If these voltages aren't correct, the fault could be in U9, U21, J1, R20, R22, soldering, or the interconnecting cable.

If the above checks are OK, use an oscilloscope to verify that data is present on U21 pin 28 and U9 pin 1 when you strike a key on your terminal. If not, the data isn't getting from your terminal to the TNC. Check J1, the cable and U9 again. Finally, be sure that your terminal actually uses levels less than -3 volts and greater than +3 volts for signal levels. 0 and +5 volts may not work, especially if they are being used direct from a computer. For example, the Commodore 64 uses TTL levels that will work only if they are inverted! This is because RS-232C logic polarity definitions are opposite those of TTL.

PACKET RADIO PROTOCOL

Explanation of Protocol

The material in this chapter is intended to supply an overview of the protocol used to transmit data by the TAPR software. References are given to more detailed information required by those wishing to implement these protocols on other hardware. The material presented below is somewhat tutorial in nature for those who have not had previous exposure to layered network protocols, but it presumes some knowledge of general communications hardware and software. Persons already well versed in networking may want to skip this chapter and refer to the primary defining document, Amateur Packet-Radio Link-Layer Protocol, AX.25 Version 2.0, available from the ARRL, 225 Main Street, Newington, CT 06111 ($8.00 US, postpaid in the United States as of this writing).

The MFJ TNC hardware and software architecture is organized in accordance with the International Standards Organization (ISO) layered network model. The model describes seven levels and is officially known as the ISO Reference Model of Open Systems Interconnection, or simply the ISO Model. The model and many other interesting topics are discussed in Computer Networks by Andrew S. Tanenbaum.

The ISO model provides for layered processes, each supplying a set of services to a higher level process. The MFJ TNC currently implements the first two layers, the Physical layer and the Data Link layer.

Physical Layer

The duty of the Physical Layer, layer one, is to provide for the transmission and reception of data at the bit level. It is concerned only with how each bit is physically transmitted, i.e., voltages on a hardwire line or modem tones on phone or RF links.

The physical layer of the MFJ TNC is described in Chapter 7, Hardware. It is compatible with the various TNCs currently available to radio Amateurs. The actual modem interface is compatible with the Bell 202 standard which is similar to the CCITT V.23 standard. Any other hardware device which is compatible with the Bell 202 standard should

be compatible with the MFJ TNC, at least at level one of the ISO reference model.

Data Link Layer

The duty of the Data Link layer is to supply an error-free stream of data to higher levels. Since level one simply passes any bits received to level two and is unaware of the content or overlying structure of the data, transmission errors are not detectable at level one. Level two carries the responsibility of detecting and rejecting bad data, retransmitting rejected data, and detecting the reception of duplicate data.

Level two accomplishes this task by partitioning data to be transferred by level one into individual frames, each with its own error detection field and frame identification fields. The MFJ TNC supports two versions of a level-two layer, AX.25 version 1.0 and AX.25 version 2.0. Each of these protocols is based on HDLC, the High-Level Data Link Control protocol defined by the ISO.

HDLC Frames

Exact knowledge of the format of HDLC frames has been made largely unnecessary by the advent of LSI and VLSI communications chips which interface directly with the level one hardware. The level two software need only supply data to fill in various fields and the chip takes care of the rest. For completeness however, an HDLC frame looks like this:

| FLAG | ADDRESS | CONTROL | PID & DATA | FCS | FLAG |

FLAG — A unique bit sequence (01111110) used to detect frame boundaries. A technique called "bit stuffing" is used to keep all other parts of the frame from looking like a flag.

ADDRESS — A field normally specifying the destination address. AX.25 uses a minimum of 14 bytes and a maximum of 70 bytes containing the actual call signs of the source, destination, and optionally up to eight digipeaters.

CONTROL — A byte which identifies the frame type. In the AX.25 protocol, the control field may include frame numbers in one or two 3-bit fields.

PID — A Protocol Identification byte appears as the first byte of the HDLC DATA field in AX.25 Level Two information frames, and identifies which Level

3 protocol is implemented, if any. In the case where no Level 3 protocol is implemented, PID = $F0.

DATA This field contains the actual information to be transferred. This field need not be present. Most frames used only for link control do not have data fields.

FCS Frame Check Sequence, a 16-bit error detection field.

The communications chip recognizes the opening and closing flags and passes the address, control, and data (including PID) fields to the software. The FCS field is a Frame Check Sequence computed by the transmitting chip and sent with the frame. The receiving chip recomputes the FCS based on the data received and rejects any frames in which the received FCS does not match the computed FCS. There is virtually no chance of an undetected bad frame using this method. This satisfies the level two task of bad data detection.

The communications chip used in the MFJ TNC 2 is a Zilog 8440 SIO operating in conjunction with a two-chip "state machine" which is used to recover the data clock. The transmitted data is encoded in NRZI form, which encodes a "0" data bit as a transition in the encoded bit stream and a "1" data bit as no transition. This, in combination with the "bit-stuffing" which ensures that no more than five "1"s occur in a row except when FLAG bytes are being transmitted, guarantees that a logic level transition occurs at least once every 5 bit times. These frequent transitions allow the receiver to synchronize its clock with the transmitter. Other chips which are compatible with the SIO + "state machine" are the Western Digital 1933/1935 (used on the TAPR TNC 1, AEA PKT-1, Heathkit HD-4040, etc.), the Intel 8273 (used on the VADCG and Ashby TNCs) and the Zilog 8530 (used on the Xerox 820 FAD adapter).

While the HDLC format supplied by the communications chips is used by the AX.25 protocol, there are several other Layer Two concerns. These are duplicate frame detection, connection and disconnection of the level two layers on different TNCs, and buffer overrun avoidance. The AX.25 protocol solves these problems as described below.

AX.25 Level Two

AX.25 is based on the Balanced Link Access Procedure (LAPB) of the CCITT X.25 standard. LAPB in turn conforms to the HDLC standard. Two extensions are made to LAPB in AX.25. These are the extended address field, and the unnumbered

information (UI) frame. In LAPB, addresses are limited to eight bits, while AX.25 uses from 112 to 560 bits, containing the originator's call sign, the destination call sign and an optional list of one to eight digipeater (simplex digital repeater) call signs.

The UI frame is used to send information bypassing the normal flow control and acknowledgment protocol. The UI frame is not acknowledged but can be transmitted at layer two without fear of disturbing higher layers. It is used for beacon frames, for automatic identification packets, and for sending information frames when the TNC is not connected to another TNC, e.g., CQ and QST activities.

The exact specifications for AX.25 are supplied in the ARRL publication Amateur Packet-Radio Link-Layer Protocol, AX.25 Version 2.0. The TAPR implementation adheres to this standard for AX.25 version 2.0. The implementation of version 1.0 is almost identical to the TAPR TNC 1 version of AX.25 protocol in software releases 3.x. This provides compatibility with the majority of Amateur packet radio stations.

The following table lists the frame types used by AX.25 and describes their purpose. This material is provided to give a general understanding of the protocol, and is not intended to replace the published specification. The byte fields are given as they appear in memory after data is received, i.e., the high order bit is at the left and the low order bit is at the right. This is also the format of the display provided by the **TRACE** command. Some texts, including the AX.25 protocol specification, list the bits in the order in which they are transmitted, which is low order bit first.

The control bytes are presented in hex with "x" used to indicate four bits which depend on the acknowledge functions the packet is performing. Usually "x" is a frame number. Frame numbers fit into three bits and are used to ensure that frames are received in order and that no frames are missed. Since only three bits are available, the frame number is counted modulo 8. This is why the **MAXFRAME** parameter has a ceiling of 7: no more than seven frames can be "in flight" (transmitted but unacknowledged) at one time. A short description of the use of the frames is given after the table.

Table 9-1. AX.25 Control Codes

Code	Abbrv	Frame Type
x1	RR	Receive Ready
x5	RNR	Receive Not Ready
x9	REJ	Reject
03	UI	Unnumbered Information
0F	DM	Disconnected mode
2F	SABM	Connect request
43	DISC	Disconnect request
63	UA	Unnumbered Acknowledge
87	FRMR	Frame reject
even	I	Any frame ending in an even number (including $A, $C, and $E) is an information frame.

I This and UI frames are the only frame types containing user data. The control byte contains this frame's number and the number of the next frame expected to be received from the other end of the link.

RR Usually used to acknowledge receipt of an I frame. The RR function can also be performed by sending an I frame with an updated "expected next frame number" field.

RNR Used when the buffer space on the receiving side is full.

REJ Used to request retransmission of frames starting from "x". Missed frames are detected by receiving a frame number different from that expected.

DM Sent in response to any frame received other than a connect request (SABM) when the TNC is disconnected. Sent in response to an SABM whenever the TNC is on the air but can't connect to the requesting user, e.g., if the TNC is already connected to someone else or if **CONOK** is **OFF**.

SABM Set Asynchronous Balanced Mode - initiates a connect.

DISC Initiates a disconnect.

UA Sent to acknowledge receipt of an SABM or DISC.

FRMR Sent when an abnormal condition occurs, i.e., the control byte received is undefined or not proper protocol at the time received.

UI An I frame without a frame number. It is not acknowledged.

Channel Use and Timing Functions

The following discussions mention timing parameters which are set by various commands. These timing functions are also discussed in Chapter 5.

An important part of any packet radio protocol is the means by which many stations make efficient use of an RF channel, achieving maximum throughput with minimum interference. The basis for this time domain multiplexing is **C**arrier-**S**ensed **M**ultiple **A**ccess (CSMA) with collision detection and collision avoidance.

CSMA means simply that (as every Amateur knows) no station will transmit if the frequency is in use. The TNC continually monitors for the presence of an audio data carrier on frequency and transmits only if there is no carrier. (The RF carrier is not normally detected; however, an input is available on the TNC 2 radio interface connector to allow such an input.) In order to make detection of a busy channel more reliable, the TNC sends an audio signal (continuous flags) any time the transmitter is keyed and a packet is not being sent, as during the transmitter keyup delay (**TXDELAY**), or while a slow audio repeater is being keyed (**AXDELAY**).

By itself, CSMA is not enough to insure a minimum, or even low, interference rate, due to the likelihood of simultaneous keyup by two or more stations. This is where collision detection and collision avoidance come in. The TNC detects a collision by the absence of an ACK from the station it is sending to. The receiving station does not acknowledge the frame that suffered the collision, since either the FCS was incorrect or the packet was not heard. There are other possible reasons for non-receipt of the packet, but the TNC's response is based on the assumption of a collision.

After transmitting a packet, the TNC waits a "reasonable" length of time (**FRACK**) for an acknowledgment. "Reasonable" is determined by the link activity, frame length, whether the packet is being digipeated, and other time-related factors. If no ACK is received, the packet must be re-sent. If the unACKed frame was lost due to a collision, the assumption is that there is at least one other packet station out there that also lost a frame and will probably have exactly the same criterion for deciding when to retry the transmission as this station is using.

In order to avoid a second collision, the collision avoidance protocol calls for the stations retrying transmissions

to wait a random time interval after hearing the frequency become clear before they key their transmitters. There must be enough different random wait times to provide a reasonable chance of two or more stations selecting different values. The difference between adjacent time values must be similar to the keyup time delay of typical stations on the frequency. This is the time lapse after a station keys its transmitter before other stations detect its presence on the channel, and is a function of the keying circuitry of the transmitter and the signal detection circuitry of the receiver. We have chosen the random time to be a multiple (0-15) of the transmitting station's keyup delay (**TXDELAY**). This is reasonable if one's own keyup delay is similar to that of other stations on the channel.

One other factor must be taken into consideration in optimizing data throughput. The currently implemented link protocols provide for relaying (digipeating) of packets. The acknowledgment procedure for such packets is that the relay station simply repeats packets without acknowledgment to the sending station. The receiving station sends its ACK back through the same digipeaters to the originating station. Since the digipeated packets are not acknowledged to the digipeater, an unsuccessful transmission must be retried from scratch by the originating station. In order to help alleviate the congestion of the frequency that tends to result when digipeated packets suffer collisions, the digipeater is given first shot at the frequency every time it becomes clear. Other stations, instead of transmitting as soon as they hear the channel clear, must wait a short time (**DWAIT**). This restriction applies to all stations except the digipeater, which is permitted to transmit relayed packets immediately. This prevents digipeated packets from suffering collisions except on transmission by the originating station.

A special time delay (**RESPTIME**) is used as the minimum wait time prior to transmitting acknowledgment frames, to prevent TNCs accepting data at high speed from the asynchronous port from colliding with acknowledgment frames when fewer than **MAXFRAME** packets are outstanding. The receiving TNC will wait long enough before sending the ACK so that it will hear the data packet which would have caused the collision, thus avoiding a fairly frequent source of delay in versions of AX.25 prior to 2.0.

Channel Flow Control

Flow control of data through the link is determined by the rate at which data is being supplied to a sending TNC and accepted from a receiving TNC.

A TNC receiving data from the link will send an RNR when the next I frame successfully received will not fit into the buffer for output to the serial port.

Whenever a TNC transmitting data received from the serial port over the link runs out of temporary buffer space, the serial port will be halted by an XOFF character or CTS signal. In the TNC 2 implementation this happens whenever there are 7 packets built and less than 210 characters left in the buffer for input from the serial port.

When the TNC receiving data from the link clears out its buffers, it sends an RR to the transmitting TNC. In order to guard against the possibility of the RR being lost and the link becoming permanently locked, the transmitting TNC will periodically re-transmit the packet that provoked the RNR. The receiving TNC will continue to respond with RNR until it can accept the packet.

BIBLIOGRAPHY

American Radio Relay League, The 1985 ARRL Handbook for the Radio Amateur, 1984, pp. 19-21 through 19-31.

American Radio Relay League. Proceedings of the First ARRL Amateur Radio Computer Networking Conference, October 16-17, 1981 (Available from ARRL for $8).

American Radio Relay League, Proceedings of the Second ARRL Amateur Radio Computer Networking Conference, March 19, 1983 (Available from ARRL for $9).

American Radio Relay League, Proceedings of the Third ARRL Amateur Radio Computer Networking Conference, April 15, 1984 (Available from ARRL for $10).

American Radio Relay League, Proceedings of the Fourth ARRL Amateur Radio Computer Networking Conference, March 30, 1985 (Available from ARRL for $10).

American Telephone and Telegraph Company, "Operations Systems Network Communications Protocol Specification BX.25," Issue 2, Publication 54001, 1979.

David W. Borden, K8MMO, and Paul L. Rinaldo, W4RI, "The Making of an Amateur Packet-Radio Network," QST, October 1981, pp. 28-30.

D. W. Davies, D. L. A. Barber, W. L. Price and C. M. Solomonides, Computer Networks and Their Protocols, John Wiley and Sons, 1979.

Exar Integrated Systems, "Phase-Locked Loop Data Book," Exar Applications Note AN-01, P. O. Box 62229, Sunnyvale, CA 94088, 1981.

Terry Fox, WB4JFI, "AX.25 Amateur Packet-Radio Link-Layer Protocol, Version 2.0," ARRL, October, 1984 (Available from ARRL for $8).

International Standards Organization, "Reference Model of Open Systems Architecture," Document ISO/TC97/SC16/N227, June 1979.

Lyle Johnson, WA7GXD, "Join the Packet-Radio Revolution," 73 Magazine, September, 1983, pp. 19-24.

Lyle Johnson, WA7GXD, "Join the Packet-Radio Revolution - Part II," 73 Magazine, October, 1983, pp. 20-31.

Lyle Johnson, WA7GXD, "Join the Packet-Radio Revolution - Part III," 73 Magazine, January, 1984, pp. 36-44.

Margaret Morrison, KV7D, and Dan Morrison, KV7B, "Designing the TAPR TNC Audio Input Filter," Proceedings of the Second ARRL Amateur Radio Computer Networking Conference, March 19, 1983, p. 54.

Margaret Morrison, KV7D, and Dan Morrison, KV7B, "Amateur Packet Radio: part 1," Ham Radio, July 1983, pp. 14-25.

Margaret Morrison, KV7D, Dan Morrison, KV7B, and Lyle Johnson, WA7GXD, "Amateur Packet Radio: part 2," Ham Radio, August 1983, pp. 18-28.

National Semiconductor, "Technical Product Information, MF10", 2900 Semiconductor Drive, Santa Clara, CA 95051.

Harold E. Price, NK6K, "Multi-use Design Considerations for the TAPR TNC," Proceedings of the Second ARRL Amateur Radio Computer Networking Conference, March 19, 1983, p. 62.

Harold Price, NK6K, "What's All This Racket About Packet?" QST, July, 1985, pp. 14-17.

Harold Price, NK6K, "A Closer Look at Packet Radio," QST, August, 1985, pp. 17-20.

Andrew S. Tanenbaum, Computer Networks, Prentice-Hall, 1981.

INDEX

$ (hex designator), 5-2, 6-3

'R, 6-31

1200 baud operation
  modem tones, 5-20

300 baud operation, 5-18
  demodulator DCD filter, 5-19, 7-6
  modem calibration, 5-19
  modem tones, 5-20

8-bit data, 5-6, 6-5, 6-6

9600 baud operation, 5-16, 7-9

<BACKSLASH>, 4-14, 6-47
<BACKSPACE>, 4-2, 4-13, 6-25
<C>, 6-35
<CR>, 4-2, 4-15, 5-6, 5-12, 6-1, 6-13
  followed by <NULL>s, 6-42
  inserting in **text** strings, 6-45
<CTRL->, 5-1
<D>, 6-35
<DELETE>, 4-13, 6-25
<DM>, 6-35
<ESCAPE>, 5-6, 6-28
<LF>, 4-5, 5-6, 5-12
  followed by <NULL>, 6-42
<NULL>, 6-42
<UA>, 6-35

?bad, 6-58
?call, 6-58
?clock not set, 6-58
?EH, 6-58
?not enough, 6-58
?not while connected, 6-58
?range, 6-58
?too long, 6-59
?too many, 6-59
?VIA, 6-59

Abbreviations, 4-3, **6-2**
Acknowledgment, 4-9, 4-12, 5-14, 5-16
  digipeated packets, 5-14
  end-to-end, 5-14
  minimum delay, 6-48
  of multiple packets, 6-30
  retries before, 6-48
Address
  packet, 4-3
  routing information, 4-7
  setting with **callsign**, 6-40
ADDRESS, HDLC field, 9-2
Analog loopback, 6-55
  full duplex operation, 6-30
Apple II, 2-2
Apple Macintosh
  serial port interfacing, 2-3
  terminal program, 2-6
ASCII codes, 5-2
Audio frequency response, 5-19
Audio repeaters, 5-13, 6-7, 6-8
Automatic operations
  beacons, 5-11
  connect, 5-11
  connect message transmission, 5-11, 6-17
  connect refusal, 5-11
  data packetizing, 5-16
  data packetizing in Converse Mode, 6-22
  digipeating, 5-10, 5-11, 6-26
  disconnect, 6-15, 6-48
  entry to data-transfer mode, 6-18, 6-41
  inclusion of send-packet character in packet, 6-23
  line wrapping, 6-49
  packet transmission, 6-44
  return to Command Mode from data-transfer mode, 6-41
  station identification, 6-31
  time stamping, 6-24
  transmit operation, 5-11, 6-55
Automatic station operation, 5-5
AX.25 Level 2 protocol, 5-12, 6-7, **9-1**

Balanced Link Access Procedure, 9-3
Battery backed-up RAM, see also bbRAM, 4-2
Baud rate
  caution notice, 2-7, 4-1
  radio port, 4-1
  serial port, 2-7, 7-1, 7-8, 8-6
  switch, 7-4
Baud-rate generator, 7-3, **7-4**
bbRAM, 4-2, 5-3, **7-2**, 7-3
  defaults, 6-57
  protection circuit, 7-5
  troubleshooting, 8-3

Beacons, **5-11**
  sent as UI frames, 9-4
  setting address, 6-54
  unnecessary transmissions, 6-9
Bell 202 standard, 9-1
BREAK signal, 4-6, 4-13, **5-5**, 5-7
Buffer sizes, 9-8
Buffers, 5-7
Bulletin board systems, 5-11, 5-16
  automatic packet transmission, 6-22
  beacon message, 6-11
  compatibility with TNC 1, 6-41
  serial port signals for, 2-1
  time stamping of connections, 6-21
  Transparent Mode, 6-52
  type-in flow control, 6-29
Busy signal, 6-20, 6-60, 6-61
  sent as DM frames, 9-5

Cables
  troubleshooting, 8-2, 8-7
Calibration
  commands, **6-12**
  formula for demodulator tones, 5-21, 6-13
  formula for modulator tones, 5-21, 6-13
  routine, 5-21
  troubleshooting, 8-4
Call sign, 4-3, **5-10**
callsign parameter, 6-3
  setting, 6-40
Cancel-line character, 4-14, 6-13
Cancel-packet character, 5-5, 6-14
Carriage return, 4-2, 6-1
Carrier-sensed multiple access, 9-6
CCITT V.23 standard, 9-1
CCITT X.25 standard, 9-3
Checksum, 6-57
Clear To Send, **2-9**, 5-8, **7-7**
Clock
  interrupt, 7-4
  MF-10 switched-capacitor filter, 7-1
  NRZ to NRZI encoder, 7-4
  processor, 7-1
  speed selection, 7-4
  user port, 7-1
Clock oscillator, 7-3
Clock recovery, 7-3
cmd: prompt, **4-1**
Cold solder joints, 8-2
Collision, 4-9, 5-14
  avoidance, 5-15, 6-29, 6-48, 9-6
  detection, 9-6
Color Computer, 2-4

Command abbreviations, 4-3, **6-2**
Command examples, 6-3
Command list, **6-5**
  format, 6-2
Command Mode, 4-1, 4-13, **5-3**, 5-9, 7-2
  <NULL> transmission, 6-43
  automatic entry, 5-4
  display features, 5-3
  display output cancellation, 6-14
  echos, 6-28
  entry from Converse Mode, 4-6, 4-13, 5-5, 6-17
  entry from Transparent Mode, 5-7
  entry to data-transfer mode, 6-41
  error messages, **6-58**
  exit to Converse Mode, 6-21
  exit to Transparent Mode, 6-52
  line cancellation, 6-13
  prompt, **4-1**, 5-3
  special characters, 5-3
Command Mode entry character, 4-13, 5-1, 5-5, 5-7, 6-17
  Transparent Mode, 6-16
Command Mode prompt, 6-57
Commands, **4-2**, **6-1**
  8BITCONV, 5-6, **6-5**
  AUTOLF, 4-5, 5-4, 5-6, **6-6**
  AWLEN, 4-4, 6-5, **6-6**
  AX25L2V2, 5-12, 5-16, **6-7**, 6-15
  AXDELAY, 5-13, 5-19, **6-7**
  AXHANG, 5-13, 5-19, **6-8**
  BEACON, 5-11, **6-9**, 6-11
  BKONDEL, 4-14, 5-4, 5-6, **6-10**
  BTEXT, 5-11, 6-9, **6-11**, 6-58
  BUDLIST, 5-17, **6-12**, 6-32
  CALIBRA, 3-3, 3-6, 5-22, **6-12**
  CALSET, 5-21, **6-13**
  CANLINE, 4-14, 5-3, 5-6, **6-13**
  CANPAC, 5-3, 5-5, 5-6, **6-14**
  CHECK, 5-12, 5-16, **6-15**
  CMDTIME, 5-7, **6-16**
  CMSG, 5-11, **6-17**, 6-23
  COMMAND, 4-13, 5-6, **6-17**
  CONMODE, 5-4, **6-18**
  CONNECT, 4-6, 4-8, 4-11, 4-12, 5-4, 5-10, 6-18, **6-19**, 6-59
  CONOK, 5-11, **6-20**, 6-60
  CONSTAMP, 5-18, **6-21**, 6-24
  CONVERS, 4-7, 5-4, **6-21**
  CPACTIME, 5-16, **6-22**, 6-49
  CR, 5-5, 5-12, 6-22, **6-23**, 6-49
  CTEXT, 5-11, 6-17, **6-23**, 6-58
  DAYTIME, 5-18, 6-21, **6-24**, 6-37, 6-39, 6-58
  DAYUSA, 5-18, 6-21, **6-25**, 6-39
  DELETE, 4-13, 5-3, 5-6, **6-25**
  DIGIPEAT, 5-11, **6-26**

DISCONNE, 4-7, 4-12, **6-26**
DISPLAY, 4-10, 6-14, **6-27**
DWAIT, 5-15, 5-19, **6-27**
ECHO, 4-4, 5-4, 5-6, **6-28**
ESCAPE, 5-6, **6-28**
FLOW, 5-4, 5-6, 5-9, **6-29**
FRACK, 5-14, **6-29**
FULLDUP, 5-19, **6-30**
HEADERLN, 4-10, **6-30**
HID, 5-10, **6-31**
ID, 5-10, **6-31**
LCALLS, 5-17, 6-12, **6-32**
LCOK, 5-4, 5-6, **6-33**
LFADD, 5-12, **6-33**
MALL, 5-17, **6-34**
MAXFRAME, 5-15, 5-18, **6-34**
MCOM, 5-18, **6-35**
MCON, 4-10, 5-17, **6-35**
MFILTER, 5-6, 5-18, **6-36**
MHCLEAR, 4-10, **6-36**
MHEARD, 4-10, **6-37**
MONITOR, 4-9, 5-17, 6-12, **6-38**
MRPT, 4-10, 6-30, **6-39**
MSTAMP, 6-24, 6-30, **6-39**
MYCALL, 4-2, 5-10, **6-40**
NEWMODE, 5-4, 6-18, **6-41**
NUCR, 4-5, 5-4, 5-6, **6-42**
NULF, 4-5, 5-4, **6-42**
NULLS, 4-5, 5-4, 5-6, **6-43**
PACLEN, 5-12, 5-15, 5-18, 6-34, **6-43**
PACTIME, 5-6, 5-12, 5-16, 6-22, **6-44**
PARITY, 4-4, **6-44**
PASS, 4-15, 5-3, 5-6, 6-11, **6-45**
PASSALL, 5-13, 6-37, **6-46**
REDISPLA, 4-14, 5-3, 5-6, **6-47**
RESET, 4-2, 5-3, **6-47**, 6-57
RESPTIME, 5-12, 5-16, **6-48**
RETRY, 4-8, 5-14, 6-19, **6-48**
SCREENLN, 4-5, 5-4, 5-6, **6-49**
SENDPAC, 5-5, 5-6, **6-49**
START, 4-14, 5-3, 5-6, 5-8, **6-50**
STOP, 4-14, 5-3, 5-6, 5-8, **6-50**
TRACE, 5-13, 6-14, **6-51**
TRANS, 5-4, **6-52**
TRFLOW, 5-9, **6-52**
TXDELAY, 4-11, 5-13, 5-19, **6-53**
TXFLOW, 5-9, **6-53**
UNPROTO, 5-13, 6-9, **6-54**
XFLOW, 5-8, 5-9, **6-54**
XMITOK, 5-11, **6-55**
XOFF, 5-3, 5-6, 5-8, **6-55**
XON, 5-3, 5-6, 5-8, **6-56**

Commodore 64 or VIC-20
  serial port interfacing, 2-3
  terminal program, 2-6
Computer interfacing, **2-1**
  Apple II series, 2-2
  Apple Macintosh, 2-3
  Commodore 64 or VIC-20, 2-3
  computers with 25-pin RS-232C ports, 2-4
  IBM PCjr, 2-3
  IBM Personal Computer, 2-2
  NEC 8201, 2-4
  non-standard serial ports, 2-5
  Radio Shack, 2-2
  Radio Shack Color Computer, 2-4
  Radio Shack Model 100, 2-4
  Sanyo MBC-55X, 2-2
Computer program, see Terminal program, **2-1**
CON LED, 2-9, 4-6, 5-22, 7-10
Connect, 4-5, **4-6**
  acknowledged by UA frames, 9-5
  automatic, 5-11
  automatic message transmission, 5-11
  errors, 6-19, 6-59
  in progress, 6-59
  initiated by SABM frames, 9-5
  message, 6-60
  request message, 6-60
  starting a QSO, 4-11
  timeout, 6-15
  unsuccessful, 4-8
Control characters, 4-6, 5-1
CONTROL, HDLC field, 9-2
Converse Mode, 4-6, 4-13, **5-5**, 5-9, 5-17, 7-2
  8-bit data, 6-5
  <NULL> transmission, 6-43
  automatic data packetizing, 5-16, 6-22, 6-44
  display features, 5-6
  display output cancellation, 6-14
  echos, 6-28
  entry from Command Mode, 5-4, 6-21, 6-41, 6-57
  escape to Command Mode, 4-6, 4-13, 6-17
  flow control to computer, 6-55, 6-56
  line cancellation, 6-13
  packet cancellation, 6-14
  send-packet character, 6-49
  special characters, 5-5
Corrections, see Input editing, 4-2
CRC checksum, 4-9
  disregarding, 6-46
CSMA, 9-6
CTS, **7-7**

Data Carrier Detect, **2-9**, 5-19
Data Communications Equipment, 2-4, 7-8
Data loss, 4-5, **5-8**, 6-42
Data Set Ready, **2-9**, **7-7**
Data Terminal Equipment, 2-4
Data Terminal Ready, **2-9**, 5-8, **7-7**
DATA, HDLC field, 9-3
Data-transfer mode, 5-2
  automatic entry, 5-4
  Converse Mode, 4-6, 5-5
  entry from Command Mode, 5-4
  Transparent Mode, 5-6
Date setting, 6-24
DCD LED, 3-3, 3-7, 4-11, 5-19
DCE, 2-4
Default values, 4-2
Default wait time, 6-27
Defaults, 5-3, **6-2**
Demodulator, 7-1
  MF-10 filter, 7-1
  troubleshooting, 8-5
Demodulator DCD filter for 300 baud operation, 5-19, 7-6
Digipeaters, 4-5
Digipeating, **4-7**, 4-12, 6-7
  adjusted retry interval, 6-29
  avoiding collisions, 6-27
  default wait time, 5-15
  display route, 6-59
  enabling and disabling, 6-26
  end-to-end acknowledgments, 5-14, 6-27, 9-7
  frame-acknowledge time, 5-14
  identification, 5-10
  monitoring route addresses, 6-39
  multiple intermediate stations, 4-8
  protocol dependence, 5-12
  specifying route, 4-8, 6-19
  station identification, 6-26, 6-31
  unconnected packets, 5-13, 6-54
Digital loopback, 4-5, 6-55
  full duplex operation, 6-30
Diodes
  incorrect installation, 8-2
DIP header
  U15, 7-6
  U17, 7-6
  U19, 7-6
DIP switch
  caution notice, 2-7, 4-1
  settings for radio baud rates, 4-1
  settings for serial port baud rates, 2-7
DISC frames, **9-5**
  monitored, 6-35
Disconnect, 4-5, **4-7**

acknowledged by UA frames, 9-5
after exceeding retry count, 6-48
causes, 6-60
in progress, 6-59
initiating a request, 6-26
message, 6-60
sent as DISC frames, 9-5
Disk accesses, 5-8
Display features
<ESCAPE> translation, 6-28
<NULL> transmission, 6-43
cancel output, 6-14
character filtering, 5-18, 6-36
Command Mode, 5-3
Converse Mode, 5-6
double typing, 4-4, 6-28
echos, 4-4, 6-28
line spacing, 4-5, 6-6, 6-33
line wrapping, 6-49
lower case translation, 6-33
overtyping, 4-5, 6-33, 6-42
redisplay line, 4-14
rubout echo, 4-14, 6-10
screen width, 4-5
Transparent Mode, 5-7, 6-52
type-in flow control, 6-29
Display format
FRMR, 6-61
monitored packets, 6-30
TRACE, 6-51
Displays (computer)
printing, 6-10, 6-42
video, 6-10, 6-42
DM frames, 6-20, 6-60, 6-61, **9-5**
monitored, 6-35
Double typing, 4-4, 6-28
DSR, **7-7**
DTE, 2-4
DTR, **7-7**

Echos, 4-4
Electrolytic capacitors
incorrect installation, 8-2
Emergency power, 5-11
End-to-end acknowledgments, 5-14
EPROM, 7-4
Error message, 6-2
Error messages, **6-58**
Error-checking
disabling, 5-13
Error-free communications, 4-5, 9-3
disabling, 6-46

FCS, HDLC field, 9-3
File transfer, 5-8, 5-8, 5-16, 6-48
  ASCII, 6-6
  binary files, 5-6, 6-6
  limiting unacknowledged data, 6-34, 6-43
  non-text files, 6-52
  serial port signals for, 2-1
Filter configurations, 5-20
Filtering received characters, 6-36
FLAG, HDLC field, 9-2
Flags, 5-13
Flow control, **5-7**
  disabling in one direction, 5-9, 6-50, 6-52, 6-53, 6-55, 6-56
  hardware, **5-9**
  link, 9-7
  packet, 9-7
  RNR frames, 9-8
  RR frames, 9-8
  software, 6-50, 6-54, 6-55, 6-56
  special characters, 5-1
  toggle character, 5-9
  Transparent Mode, 5-9, 6-52, 6-53
  type-in, **5-9**, 6-29
  XON/XOFF, **5-8**
FM operation, 3-2
Frame acknowledgment timeout, 6-29
Frame Ground, **2-8**
Framing errors, 4-4
FRMR frames, 6-61, **9-5**
FRMR in progress, 6-60
FRMR rcvd, 6-61
FRMR sent, 6-61
Full break-in CW, 6-22
Full duplex operation, 5-19, 6-30, 7-2

Gibberish, 2-8, 4-4, 8-6
Ground, 3-1
  signal location, 8-1

Hardware description, **7-3**
Hardware flow control, 5-8, **5-9**
HDLC
  frame format, **9-2**
  port, 7-5
Hex, 4-13, **5-2**, **6-3**
HF operation, 4-9, 5-16, **5-18**, 5-20, 7-12
Hum, 3-3

I frames, 6-35, **9-5**
IBM PCjr
  serial port interfacing, 2-3
  terminal program, 2-6
IBM Personal Computer, 2-2

Identification
  automatic, 5-10
  station, 5-10
Identification packet, 5-10, 6-31
  callsign text, 6-40
Identification packets, 6-31
IF bandwidth, 5-24
Immediate commands, **6-2**
Input audio filter
  for 300 baud operation, 5-19
Input editing, 6-1
  character deletion, 4-2, 4-13, 6-25
  disabled in Transparent Mode, 6-52
  line cancellation, 4-14, 6-13
  packet cancellation, 5-5, 6-14
  pass character, 6-45
  redisplay line, 4-14, 6-47
  special characters, 5-1
Integrated circuits
  incorrect installation, 8-2
  normal temperatures, 8-2
Interfacing
  computer or terminal, **2-1**
  Radio, **3-1**
International Standards Organization, 9-1
Interrupts, 5-8

J1 serial port connector, 7-7
J2 radio port connector, 3-1
J3 tuning indicator interface, **7-12**
J4 modem disconnect, 5-20, 7-4, **7-9**
  connector installation, 7-9
JMP1 DCD override, 2-9, 7-8
JMP2 clock speed selection, 7-4, 7-8
JMP4 watch-dog timer override, 3-3, 3-6, 5-22, 7-5, 7-8
JMP5 Lithium battery disconnect, 7-4
JMP6 ROM memory size selection, 7-8
JMP7 analog loopback, 3-6, 4-5, 7-8
JMP8 demodulator calibration, 5-23, 7-9
JMP9 modulator calibration, 5-22, 5-23, 7-9
JMP9 state machine bypass, 7-9
JMP10 digital loopback, 4-5, 4-11, 7-9
JMP11 transmit NRZ/NRZI selection, 7-9
JMP12 RAM size selection, 7-9
Jumper functions, **7-8**

Keyup delay, 5-13
Keywords, 6-1
  AFTER and **EVERY**, 5-12, **6-2**, 6-9, 6-44
  ASYNC, 6-27
  CHARACTE, 6-27
  CONVERS and **TRANS**, 5-4, 6-18
  ID, 6-27

LINK, 6-27
MONITOR, 6-27
NO and **YES**, **6-2**
OFF and **ON**, 6-2
TIMING, 6-27
VIA, 6-19

LAPB, 9-3
LED normal power-up display, 8-3
Line spacing of packet text, 5-12
Line wrapping, 6-49
Linefeed, 4-5
Link status messages, 5-7, 6-59
Logic lines
troubleshooting, 8-4
Lower case characters
translation to upper case, 6-33

Macintosh
serial port interfacing, 2-3
terminal program, 2-6
Memory, 7-1, 7-3
Memory space decoder, 7-3, 7-4
Messages, **6-57**
error, 6-58
link status, 6-59
MF-10 switched capacitor filter, 7-1, **7-6**
clock, 7-1
Microphone audio, 3-1
Microphone gain adjustment, 3-3
Microprocessor, 7-1, 7-3
Modem, **7-1**, 7-3, **7-6**
calibration, **5-20**
calibration for 300 baud operation, 5-19
calibration parameters, 5-24
demodulator loop timing components, 5-23
demodulator modification, 7-6
external, 5-16, 5-20, 7-2, 7-9
frequency calibration, 3-1
J4 disconnect, **7-9**
level calibration, 3-3, 3-6
MF-10 filter modification, 7-6
modification, 7-2
modulator modification, 7-6
negative power supply, 7-7
radio, 5-16
telephone, 2-1, 2-4, 2-6
troubleshooting, 8-4
Modulation level adjustment, 3-6
Modulator, 7-2
Monitoring, **4-9**, 4-13, 5-17, **6-38**
callsign list, 6-32
character filtering, 6-36

clearing list of stations heard, 6-36
connected packets, 6-34
digipeater list, 6-39
display format, 4-10, 6-30
list of stations heard, 4-10, 6-37
protocol control packets, 5-18, 6-35
routing information, 4-10
selected stations, 5-17
time stamping, 6-39
while connected, 4-10, 6-35
Multiple packets
for 1200 baud, 5-15
for 300 baud, 5-16

n parameter, 6-3
Narrow shift, 5-19
NEC 8201
serial port interfacing, 2-4
terminal program, 2-7
Net, 5-17
New lines, 4-5
Noise
audio, 3-1, 3-3
cause of retransmissions, 4-9
RF, 3-1, 3-2
Non-ASCII character sets, 6-5
NRZ to NRZI encoder, **7-5**
clock, 7-4
NRZI encoding, 9-3
Null modem cable, 7-8
Numeric values, 6-3

Operating modes, **5-2**, 7-2
OSCAR operation, **5-18**, 7-9, 7-12
full duplex, 6-30
Oscillator, **7-3**
troubleshooting, 8-3
Outstanding packets, 6-34

Packet
maximum length, 5-12, 6-43
operation, **5-10**
protocol, 5-12
re-transmission, 6-48
timing, 5-14
Packet reception
troubleshooting, 8-5
Packet Status Register, 1-2
Packet transmission
troubleshooting, 8-5
Parameters, 4-2, 6-1, **6-2**
callsign, 6-3
current value, 4-3

lists, 6-3
n, 6-3
previous value, 4-3, 4-13
text, 6-3
Parameters lists, 6-59
Parity, 4-4
even, 4-4
odd, 4-4
space, 4-4
transmission in packets, 6-44
Pass character, 4-15, 6-11, 6-23
PID, AX.25 field, 9-2
Polling, 5-8
Power requirements, 7-2
Power supply, 7-3, **7-6**
negative voltage, 8-2
troubleshooting, 8-1
Power-on reset, 7-6
LED display, 8-3
Protocol, 5-12, 5-16, **9-1**
AX.25 Level 2, 7-2, 9-1
compatibility, 7-2
control byte format, 9-4
data link layer, 9-1, 9-2
debugging function, 6-51
errors, 6-60, 6-61
HDLC frames, 9-2
HDLC implementation, 7-5
monitoring control packets, 5-18
problems, 5-13
Protocol errors
result in FRMR frames, 9-5
Push-to-talk, 3-1

R(S) adjustment, 3-6
R57, 3-5
R76 tone level adjustment, 3-3, 3-6, 7-6, 8-5
R77 modulator tone adjustment, 5-21, 7-6
R78 modulator tone adjustment, 5-22, 7-6
R79 demodulator adjustment, 5-23, 7-6
Radio baud rate, 5-8, **5-15**
optimum, 5-19
switch, 7-4
Radio interfacing, **3-1**
using accessory jack or interface box, **3-4**
using microphone connector or auxiliary jack, **3-2**
Radio port, 7-5
Radio Shack Color Computer
serial port interfacing, 2-4
terminal program, 2-7
Radio Shack computers, 2-2

Radio Shack Model 100
  serial port interfacing, 2-4
  terminal program, 2-7
Radio signals
  Ground, 3-1
  Microphone audio, 3-1
  Push-to-talk, 3-1
  Receive audio, 3-1
  Squelch input, 3-1
RAM memory, 7-1, 7-3, **7-4**
Random Access Memory, 7-3
Random wait, 5-15, 6-29
Read Only Memory, 7-3
Real-time clock, 5-18
  error message, 6-58
  setting, 6-24
Receive audio, 3-1
Receive Data, 2-2, **2-9**
Redisplay-line character, 4-14, 6-10, 6-25, 6-47, 6-57
REJ frames, **9-5**
Resuming output, 5-7
Retransmission of packets, 5-14
Retry count message, 6-61
RF loop-back, 4-12
RNR frames, **9-5**, 9-8
ROM memory, 7-1, 7-3, **7-4**
ROM, state machine, 7-5
Round-table, 5-17
RR frames, **9-5**, 9-8
RS-232C, **2-1**, 5-8
  baud rate, 2-7
  handshaking, **7-7**
  negative power supply, 7-7
  signal levels, 2-5
  signals, **2-8**
  troubleshooting, **8-6**
RS-232C signals
  Clear To Send, 2-9, 5-8
  Data Carrier Detect, 2-9
  Data Set Ready, 2-9
  Data Terminal Ready, 2-9, 5-8
  Frame Ground, 2-8
  Receive Data, 2-2, 2-9
  Signal Ground, 2-2, 2-9
  Transmit Data, 2-2, 2-9
RS-422, 2-3

SABM frames, **9-5**
  monitored, 6-35
Sanyo MBC-55X, 2-2
Secondary station ID, **5-10**

Send-packet character, 5-5, 6-13, 6-14, 6-49
  automatic inclusion in packet, 6-23
  set to <CR>, 5-12

Serial port, **2-1**, **7-1**, 7-3, **7-5**
  adapter card, 2-2
  baud rate, 2-7, 5-8, 7-8
  clock, 7-1
  connector, 2-1
  hardware handshaking, 2-8
  jumpers for auxiliary signals, 2-5
  non-standard, 7-7
  parameter configuration, 4-3
  parity, 4-4, 6-44
  signals, 2-1, **2-8**, 8-7
  troubleshooting, **8-5**
  word length, 4-4, 6-6
Serial port baud rate, 5-8
Serial port problems
  baud rate, 8-6
  data signals, 8-6, 8-7
  handshaking signals, 8-6
  parity, 8-6
  switch, 7-4
  won't accept data, 8-7
Sign-on message, **4-1**, 6-57
  appears as gibberish, 8-6
  doesn't appear, 8-6
Signal Ground, 2-2, **2-9**
Single sideband, see SSB operation, 3-2
Software flow control, **5-8**
Software UART, 5-10, 6-29
  Color Computer, 2-7
Software, see Terminal program, 2-1
Special characters, 4-13, 5-1
  Command Mode, 5-3
  Converse Mode, 5-5
  Transparent Mode, 5-6
Specifications, **7-1**
Squelch input, 3-1
SSB operation, 3-2, 5-19, 7-1
SSID, **5-10**, 6-3
STA LED, 5-22, 7-11
State machine, 7-5
Station identification, **5-10**, 6-31
  callsign text, 6-40
  sent as UI frames, 9-4
Stations heard
  clearing list, 6-36
  displaying list, 6-37
Stopping output, 4-14, 5-7
Sub-station ID, 6-3
SW2, baud rate selection, 7-4

TAPR, **1-1**
Temperature of components, 8-1, 8-2
Terminal Node Controller, **1-1**
Terminal program, 2-1, **2-6**, 4-6, 5-8
  Apple Macintosh, 2-6
  Commodore 64 or VIC-20, 2-6
  IBM PCjr, 2-6
  NEC 8201, 2-7
  Radio Shack Color Computer, 2-7
  Radio Shack Model 100, 2-7
Terminal, see also Computer, **2-1**
Test equipment, 3-2
text parameter, 6-3
  maximum length, 6-11, 6-23, 6-58
Time setting, 6-24
Time stamping, 5-18, 6-21, 6-24
  list of stations heard, 6-37
  monitored packets, 6-39
Timing, **5-13**, **9-6**
  collision avoidance, 9-6
  default wait, 5-15, 9-7
  frame-acknowledge time for digipeated packets, 5-14
  minimum acknowledgment delay, 6-48, 9-7
  packet retransmission, 5-14, 9-6
  packet transmission, 6-27
  protocol, 6-15
  random wait on retransmission, 5-15, 9-6
  repeater hang delay, 6-8
  retry interval, 6-29
  transmitter keyup, 5-13, 6-7, 6-53, 9-6
  Transparent Mode escape, 6-16
TNC, **1-1**
TNC 1
  AX.25 Level 2 version, 6-7
  AXHANG parameter, 6-8
  Command Mode entry and exit, 6-41
  protocol compatibility, 9-3, 9-4
  TXDELAY parameter, 6-53
TNC 1 **AXDELAY** parameter, 6-7
TNC Restart character, 6-53, 6-54, 6-56
TNC Stop character, 6-53, 6-54, 6-55
too many packets outstanding, 6-57
Transmit
  inhibiting, 5-11, 6-55
  watch-dog timer, 7-5
Transmit Data, 2-2, **2-9**
Transmit keying
  troubleshooting, 8-5
Transmit timing, 5-13
Transmit-receive switching, 5-13
Transparent Mode, **5-6**, 5-9, 5-16, 7-2
  automatic packet transmission, 6-44
  display features, 5-7

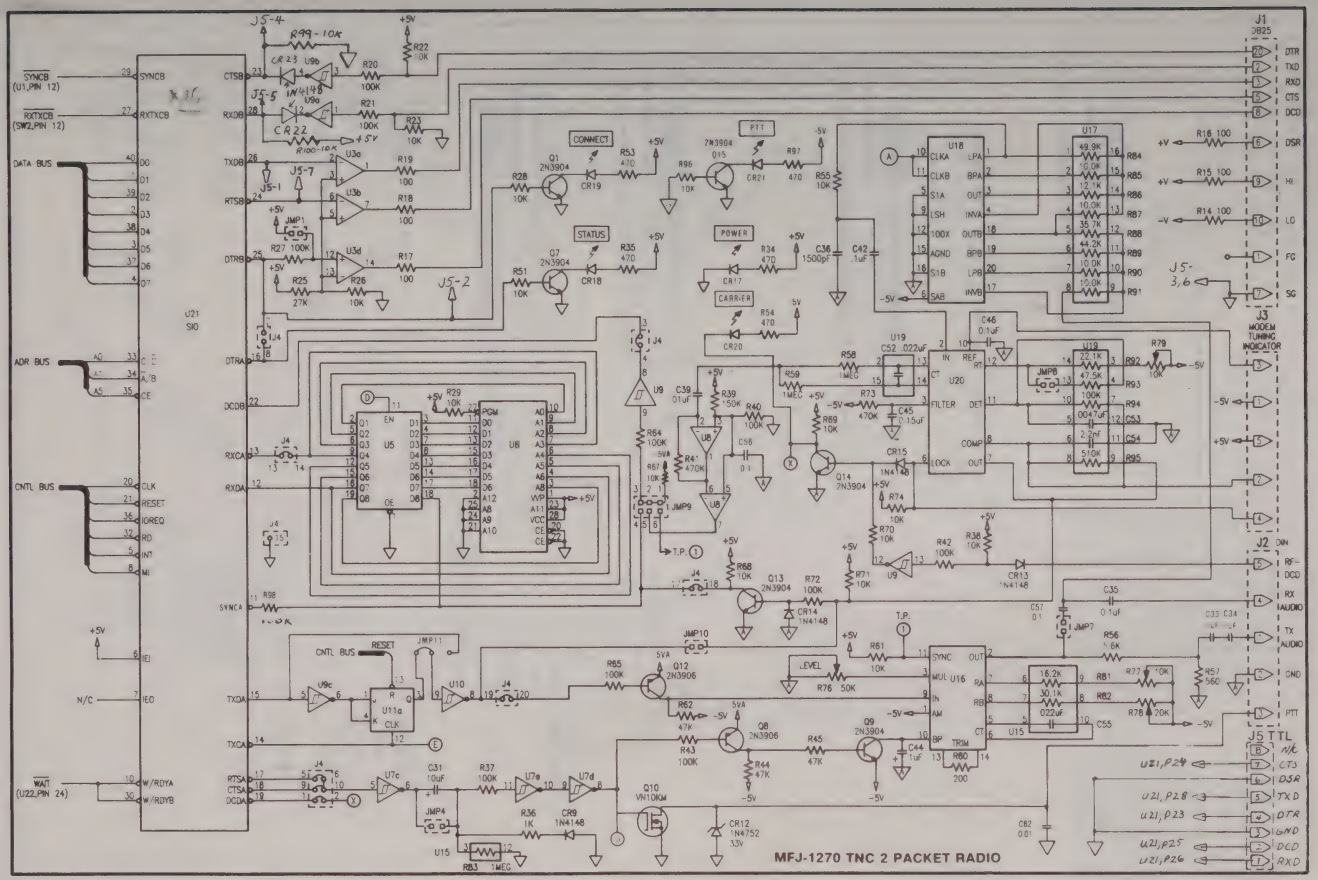
entry from Command Mode, 5-4, 6-41, 6-52, 6-57
escape timing, 6-16
flow control, 5-9
packet formatting, 5-12
software flow control, 6-52, 6-53
special characters, 5-6
Trimpots
adjustment, 5-21
Troubleshooting, **8-1**
equipment, 8-1
TTL logic levels, 8-3
Tucson Amateur Packet Radio, **1-1**
Tuning indicator, 7-2
J3 interface, 7-12
Type-in flow control, **5-9**

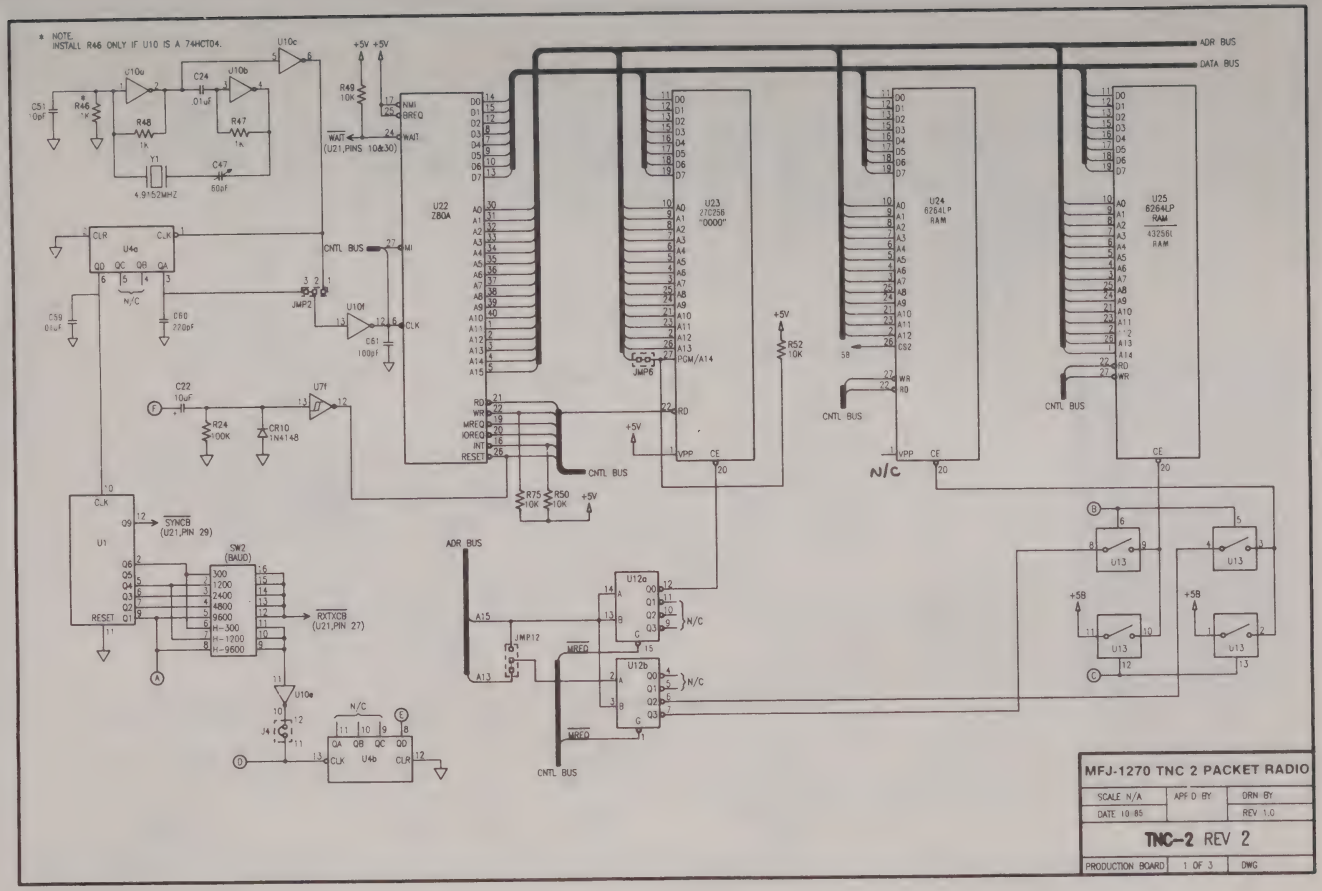
U15 modulator configuration header, 7-6
U17 MF-10 configuration header, 7-6
U19 demodulator configuration header, 7-6
UA frames, **9-5**
monitored, 6-35
UI frames, 6-31, 6-54, 9-4, 9-5, **9-6**
Unacknowledged packets, 6-34
Unconnected packets, 5-13
sent as UI frames, 9-4, 9-6
setting address, 6-54
User Restart character, 6-50, 6-52
User Stop character, 6-50, 6-52

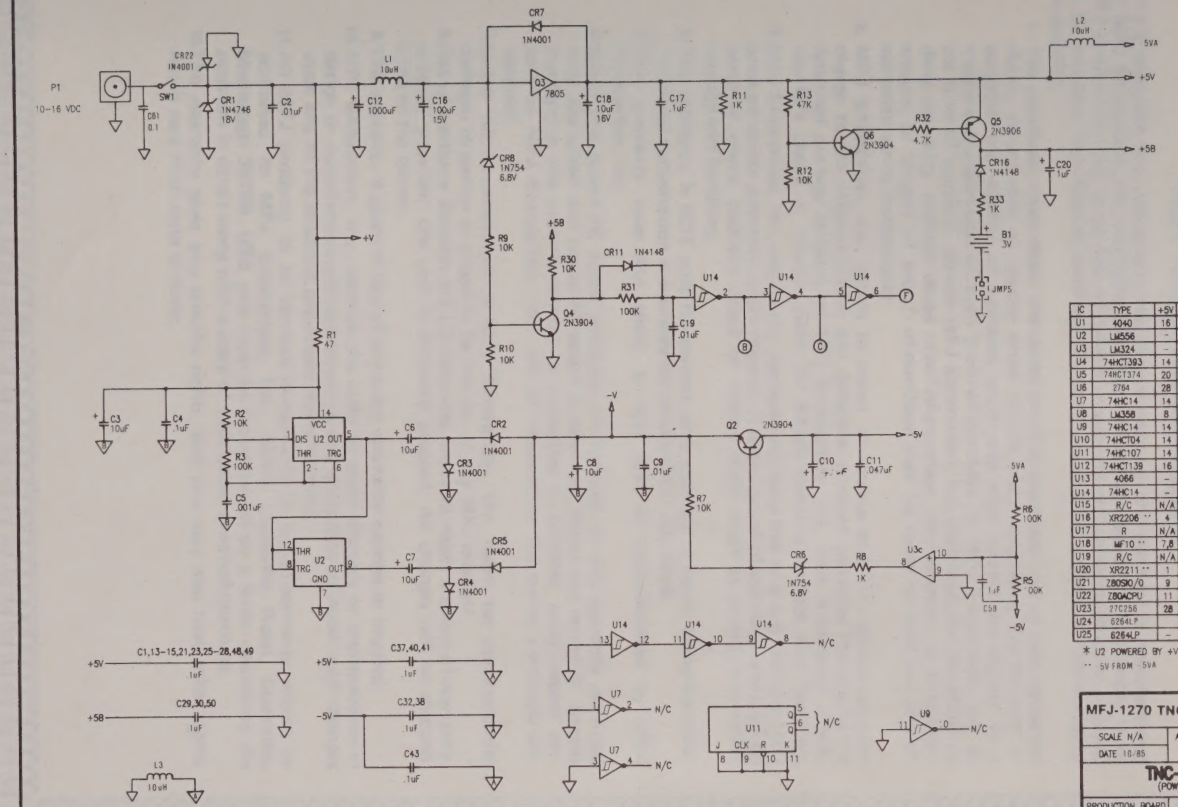
VIC-20
serial port interfacing, 2-3
terminal program, 2-6
Voice operation, 3-1
Voltage inverter, 7-3
Voltage regulators, 8-1
Volume adjustment, 3-3, 3-7

was, 6-57
Watch-dog timer, 7-3, 7-5
override, 3-3, 3-6
Watch-dog timer modification, 7-6
Word length, 4-4, 6-6

XON/XOFF flow control, **5-8**
XR2206 modulator, 7-6
XR2211 demodulator, 7-6


MFJ-1270 TNC 2 PACKET RADIO
SYNCB
RXTXCB
DATA BUS
ADR BUS
CNTL BUS
WAIT
U21 SIO
U5 U6 U7 U8 U9 U10 U11 U12 U13 U14 U15 U16 U17 U18 U19 U20
Q1 Q7 Q8 Q9 Q10 Q12 Q13 Q14
CR12 CR13 CR14 CR15 CR17 CR18 CR19 CR20 CR21 CR22 CR23
CONNECT
STATUS
POWER
CARRIER
PTT
RESET
LEVEL
TRIM
J1 DB25
J2
J3 MODEM TUNING INDICATOR
J5 TTL
DTR
TXD
RXD
CTS
DCD
DSR
RI
LO
FG
SG
DIN
RF-DCD
RX AUDIO
TX AUDIO
GND
PTT
U21,P24
U21,P2.8
U21,P23
U21,P25
U21,P26


* NOTE
INSTALL R46 ONLY IF U10 IS A 74HCT04.
ADR BUS
DATA BUS
U10c
U10a
U10b
C24
.01uF
C51
10pF
R46
1K
R48
1K
R47
1K
Y1
4.9152MHZ
C47
60pF
+5V +5V
R49
10K
NMI
BREQ
WAIT
WAIT (U21, PINS 10&30)
U22
Z80A
CNTL BUS
MI
U4a
CLR
CLK
QD QC QB QA
N/C
JMP2
C59
.01uF
C60
220pF
U10f
C61
100pF
CLK
C22
10uF
U7f
R24
100K
CR10
1N4148
RD
WR
MREQ
IOREQ
INT
RESET
CNTL BUS
U23
27C256
"0000"
JMP6
PGM/A14
U24
6264LP
RAM
U25
6264LP
RAM
43256
RAM
+5V
R52
10K
5B
WR
RD
CS2
VPP
CE
20
N/C
R75
10K
R50
10K
+5V
U1
CLK
Q9
SYNCB
(U21, PIN 29)
Q6
Q5
Q4
Q3
Q2
Q1
RESET
SW2
(BAUD)
300
1200
2400
4800
9600
Hi-300
Hi-1200
Hi-9600
RXTXCB
(U21, PIN 27)
U10e
J4
N/C
U4b
QA QB QC QD
CLK
CLR
ADR BUS
A15
A13
JMP12
U12a
U12b
MREQ
N/C
CNTL BUS
U13
+5B
MFJ-1270 TNC 2 PACKET RADIO
SCALE N/A
APP'D BY
DRN BY
DATE 10/85
REV 1.0
TNC-2 REV 2
PRODUCTION BOARD
1 OF 3
DWG

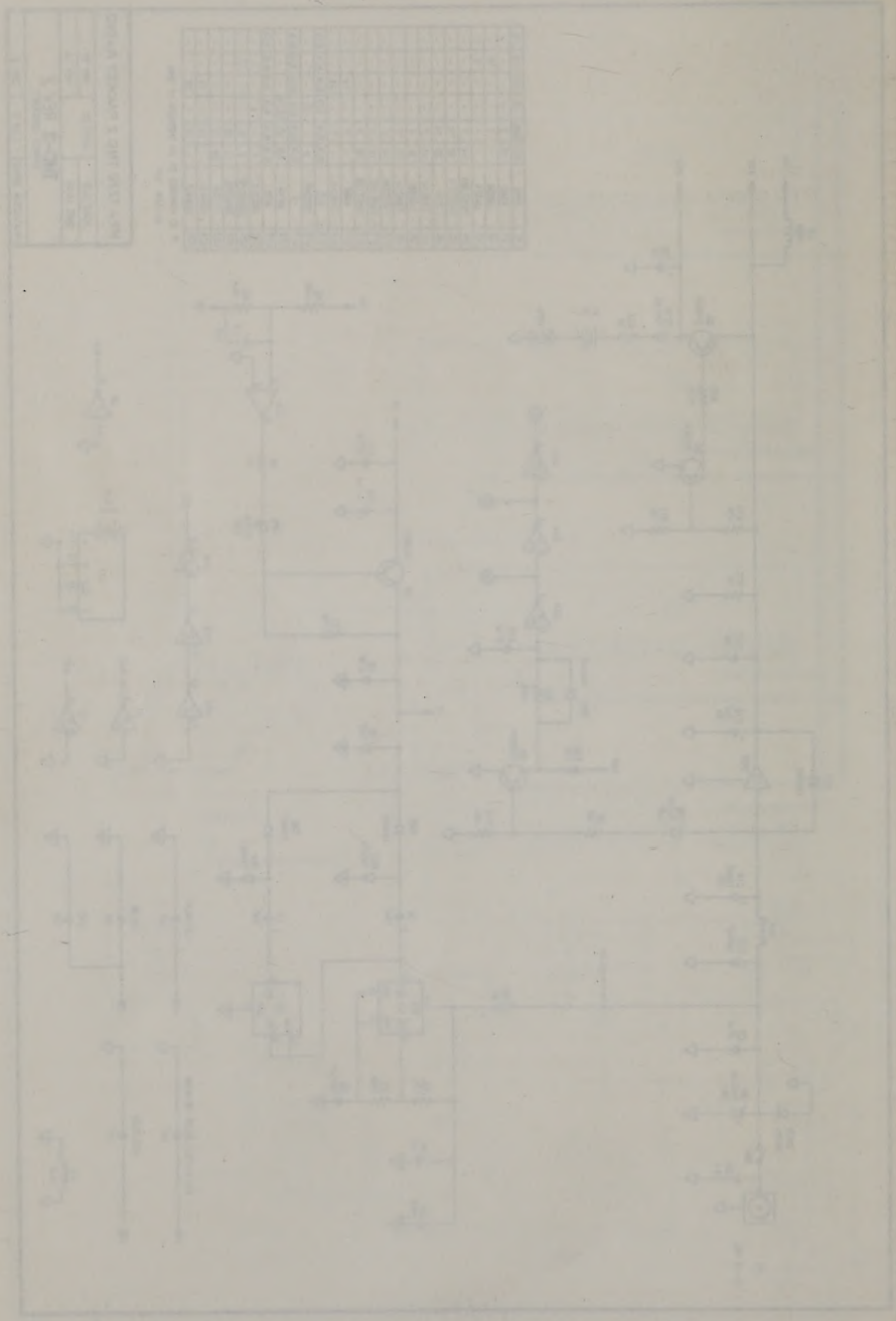


IC	TYPE	+5V	GND	-5V	+5B	+V	-V
U1	4040	16	8	–	–	–	–
U2	LM556	–	7	–	–	*	–
U3	LM324	–	–	–	–	4	11
U4	74HCT393	14	7	–	–	–	–
U5	74HCT374	20	10	–	–	–	–
U6	2764	28	14	–	–	–	–
U7	74HC14	14	7	–	–	–	–
U8	LM358	8	4	–	–	–	–
U9	74HC14	14	7	–	–	–	–
U10	74HCT04	14	7	–	–	–	–
U11	74HC107	14	7	–	–	–	–
U12	74HCT139	16	8	–	–	–	–
U13	4066	–	7	–	14	–	–
U14	74HC14	–	7	–	14	–	–
U15	R/C	N/A	N/A	N/A	N/A	N/A	N/A
U16	XR2206 **	4	–	12	–	–	–
U17	R	N/A	N/A	N/A	N/A	N/A	N/A
U18	MF10 **	7,8	15	13,14	–	–	–
U19	R/C	N/A	N/A	N/A	N/A	N/A	N/A
U20	XR2211 **	1	–	4	–	–	–
U21	Z80SIO/0	9	31	–	–	–	–
U22	Z80ACPU	11	29	–	–	–	–
U23	27C256	28	14	–	–	–	–
U24	6264LP		14	–	28	–	–
U25	6264LP	–	14	–	28	–	–

* U2 POWERED BY +V THROUGH 47 OHM.

** 5V FROM -5VA

MFJ-1270 TNC 2 PACKET RADIO		
SCALE N/A	APP'D BY	DRN BY
DATE 10/85		REV 1.0
TNC-2 REV 2 (POWER CIRCUITRY)		
PRODUCTION BOARD	3 OF 3	DWG #



FULL 12 MONTHS WARRANTY

MFJ Enterprises, Inc. warrants to the original owner of this product, if manufactured by MFJ Enterprises, Inc. and purchased from an authorized dealer or directly from MFJ Enterprises, Inc. to be free of defects in material and workmanship for a period of 12 monthes from date of purchase provided the following terms of this warranty are satisfied

1. The purchaser must retain the dated proof-of-purchase (bill of sale, cancelled check, credit card or money order receipt, etc.) describing the product to establish the validity of warranty claim and must submit the original or a machine-reproduction of such proof-of-purchase to MFJ Enterprises, Inc. at the time of warranty service MFJ Enterprises, Inc. shall have the discretion to deny warranty without dated proof-of-purchase. Any evidence of alteration, erasure, or forgery of proof- of-purchase shall be cause to void any and all warranty terms immediately.
2. MFJ Enterprises, Inc. agrees to repair or replace at MFJ's option without charge to the original owner any defective product provided the product is returned postage prepaid to MFJ Enterprises, Inc. with a personal check, cashier's check or money order for **$4.00** covering postage and handling.
3. MFJ Enterprises, Inc. will supply replacement parts free of charge for any MFJ product under warranty upon request. A dated proof-of-purchase and a **$4.00** personal check, cashier's check or money order must be provided to cover postage and handling.
4. This warranty is **NOT** void for owners who attempt to repair defective units. Technical consultation is available by calling (601) 323-5869.
5. This warranty does not apply to kits sold or manufactured by MFJ Enterprises, Inc.
6. Wired and tested PC board products are covered by this warranty provided **only the wired and tested PC board is returned.** Wired and tested PC boards installed in the owner's cabinet or connected to switches, jacks, cables etc. sent to MFJ Enterprises, Inc. will be returned at the owner's expense unrepaired.
7. Under no circumstances is MFJ Enterprises, Inc. liable for consequential damages to person or property by the use of any MFJ product.
8. **Out-of-warranty Service:** MFJ Enterprises, Inc. will repair any out-of-warranty product provided the unit is delivered prepaid. All charges will be shipped COD to the owner.
9. This warranty is given in lieu of any other warranty express or implied.
10. MFJ Enterprises, Inc. reserves the right to make changes or improvement in design or manufacture without incurring any obligation to install such changes upon any of the products previously manufactured.
11. All MFJ products to be serviced in-warranty or out-of-warranty should be addressed to **MFJ Enterprises, Inc., 921A Louisville Road, Starkville, Mississippi 39759, USA** and must be accompanied by a letter describing the problem in detail along with a copy of your dated proof-of-purchase.
12. This warranty gives you specific rights, and you may also have other rights which vary from state to state.